Technical University of Denmark
Department of Management Engineering

Master Thesis

Identifying and mitigating bias in machine learning models

June 25, 2021

Students:
Daniel Juhász Vigild, s161749
Lau Johansson, s164512

Supervisor:
Aasa Feragen

Table of Contents

1 Abstract

This thesis examines bias in machine learning models used for decision support by identifying and mitigating bias in the AIR (AI Rehabilitation) project from Aalborg Municipality. The AIR project explores the possibility of using citizens' data to predict their risk of falling and thereby support decisions regarding the provision of fall prevention training. The thesis utilizes a data set with 2144 records that contains information regarding citizens from Aalborg Municipality and their fall incidents. Bias is identified by training five machine learning models on the AIR data set, estimating gender-specific classification rates, and assessing the relation of the gender-specific rates. Bias is mitigated by applying four different pre-processing techniques (*dropping gender*, *gender swap*, *disparate impact removal*, and *learning fair representations*) on the AIR data set and re-estimating the gender-specific classification rates. Gender bias is identified in all models built on the original AIR data set. Most noticeably, the results show that the models misclassify females who fall at a higher rate than males who fall (false negatives). All four pre-processing techniques successfully mitigate the identified gender bias. For future work on bias mitigation, the thesis recommends that the applied identification and mitigation techniques could be a part of a two-step system, where the pre-processing mitigation efforts are separated from the process of building classifiers.

2 Introduction

In 2019, the Danish government published *"National strategi for kunstig intelligens"*, which describes the government's ambition for responsible development and use of artificial intelligence (AI). The technology is evolving rapidly, for example, 3 % of Danish municipalities used AI for solving tasks in 2018 - and 55 % of the municipalities expect to implement AI before the end of 2021 [1, p. 52]. In the public sector, AI can support employees in solving tasks, making decisions and give citizens a smarter and better user experience. However, the government notes that rapid evolution of technology can make citizens feel insecure about the future. They further state that algorithms must ensure equal treatment and be objective. The Danish government has developed six ethical principles for setting a common framework for developing and using AI [1, pp. 27-29]. One of the six principles for artificial intelligence is "Equality and Justice":

> *"Kunstig intelligens må ikke reproducere fordomme, der marginaliserer befolkningsgrupper. Der skal arbejdes aktivt for at forhindre uønsket bias og fremme designs, der undgår kategorisering, som diskriminerer på baggrund af fx etnicitet, seksualitet og køn. Demografisk og faglig diversitet bør være en rettesnor i arbejdet med kunstig intelligens."*
>
> [Artificial intelligence must not reproduce prejudices that marginalize population groups. Efforts must be made to prevent unwanted bias and promote designs that avoid categorizations that discriminate on e.g., ethnicity, sexuality, and gender. Demographic and professional diversity should be a guideline in the work on artificial intelligence.] [1, p. 58]

Through *"National strategi for kunstig intelligens"*, the government initiates signature AI projects in the public sector to accommodate the lack of experience related to the use of AI. To finance the signature projects, the Danish government has granted 60 Mio. DKK for 2019-2029, which complements 295 Mio. DKK prioritized in the Finance Act [1, p. 20]. The signature projects will test technology in areas where there is a potential to increase quality and productivity in core public tasks [1, p. 54].

One of the above mentioned signature projects, the **AIR (AI Rehabilitation) project**, will be the focal point of analysis in our thesis. The AIR project is owned by Aalborg Municipality, with Aarhus University and the company DigiRehab as collaborators. The project pursues the possibilities of using citizens' data regarding registered aids to calculate probabilities of falling, and through this, support decisions on who should be offered fall prevention training [2].

We will analyse the data and algorithm used in the AIR project. Our analysis focuses on the identification and mitigation of bias. In this sense, we adhere to the words of the principle of "Equality and Justice" from *"National strategi for kunstig intelligens"*. By testing techniques for identification and mitigation of bias in the AIR project, the thesis will attempt to contribute to the general field of research on bias in AI.

Bias in machine learning
To show why biased machine learning models could be problematic, we briefly present three cases of bias in machine learning.

Bias in natural language processing
When modeling human language in a machine learning context, typically called natural language processing (NLP), researchers have discovered that bias and prejudices related to protected groups (e.g. gender or race) can be identified in the mathematical representations of language learned by models [3]. Latanya Sweeney [4] showed that ads on search results for names typically associated with African-Americans were more likely to suggest criminal activity than names associated with Caucasians. In this example, the NLP algorithms used by Google reproduce bias and prejudice found in society.

Bias in facial recognition systems
The research study, *"Gender Shades"*, evaluated three commercial gender classification systems by Microsoft, IBM, and Face++. The study shows that dark-skinned females are the most misclassified group with accuracy as low as 65.3%, compared to accuracy for light-skinned males of up to 100% [5]. The authors suggest that the disparity in accuracy is due to an over-representation of lighter-skinned subjects within the data sets. If directly implemented in commercial products that use facial recognition

(e.g. phones, security systems), the performance of these would be substantially worse for dark-skinned females.

Bias in risk assessment tools
In an alleged attempt to get defendants through the US legal system as efficiently as possible, the courtrooms have turned to predictive risk assessment tools [6]. One tool for such risk assessment is the software program called Correctional Offender Management Profiling for Alternative Sanctions (COMPAS). This tool predicts a recidivism score: a number estimating the likelihood of re-offending. The judges use the risk assessment as a decision support tool when determining pretrial release and sentencing [6]. In a 2016 investigation of the COMPAS algorithm, ProPublica criticized the predictive tool for yielding biased results by showing how *"(...) blacks are almost twice as likely as whites to be labeled a higher risk but not actually re-offend"* [7]. This sparked a public debate regarding racial bias in algorithms used in the judicial system [8].

3 Research Questions

As stated in the introduction, we intend to test techniques for identification and mitigation of bias in the AIR project. Therefore, we aim to answer the following research questions in our thesis:

- *RQ1: How can bias be identified in the AIR project?*
- *RQ2: How can bias be mitigated in the AIR project?*

By answering the research questions, we intend to provide Aalborg Municipality and collaborators with actionable recommendations that can be used to improve the classifications of the AIR algorithm in terms of bias. Our vision is that the result of our thesis can be helpful to practitioners in the public sector in their work on bias identification and mitigation. In this light, we prioritize the accessibility and reproducibility of our models and implementations. Therefore, we will use standard implementations of algorithms and test a range of simple techniques to identify and mitigate bias. Furthermore, we envision that these techniques come from pre-existing libraries or are easy-to-use methods so practitioners in the public sector can apply the techniques presented in our thesis.

In section 10, we answer research question 1 by assessing the classification rates of model predictions.

In section 11, we answer research question 2 by testing four different techniques for mitigating bias.

4 Terminology

In the following section, we clarify some central concepts that will be applied in the thesis. Initially, we describe what is meant by the term *machine learning model*. Then, we introduce metrics used for measuring performance of classifiers. Finally, a short introduction to bias in machine learning is provided.

4.1 Machine learning model

Machine learning is a subfield of AI where models can learn structures in data. When learning, the model finds a mapping from input to output that is not explicitly programmed. The models can then be applied on new data where the output can be utilized by a user [9, p. 197]. Machine learning models can be used for many purposes, but in the context of our thesis, we examine the role of machine learning models when used as decision support tools [10].

The terms *AI model*, *machine learning algorithms*, *machine learning model*, and *algorithm* are used interchangeably, and all refer to the same thing unless it is clearly specified.

4.2 Classification rates

According to Verma and Rubin [11], who have made a collection of definitions of bias related to algorithmic classification problems, most statistical measures of bias use a combination of the definitions from table 1. In machine learning, the term *classification* refers to the process of developing a model that can predict categorical class labels. The confusion matrix in table 1 categorizes the predictions of a model. In a binary classification setting, the class label can take two possible values: positive class or negative class. The number of positive and negative observations that are correctly classified are called true positives (TP) and true negatives (TN), respectively. Misclassified observations are called false positives (FP) and false negatives (FN). If the class labels are, for example, boolean, then 1 or *true* labeled observations are assigned as the positive class [9, p. 203-204].

		True Class	
		P	N
Predicted Class	P	True Positives	False Positives
	N	False Negatives	True Negatives

Table 1: Confusion matrix

From the confusion matrix in Table 1, several metrics can be calculated where the most common are [12, p. 862][13]:

The true positive rate (TPR):

$$TPR = \frac{TP}{TP + FN} \tag{1}$$

The false positive rate (FPR):

$$FPR = \frac{FP}{FP + TN} \tag{2}$$

The true negative rate (TNR):

$$TNR = \frac{TN}{TN + FP} \tag{3}$$

The false negative rate (FNR):

$$FNR = \frac{FN}{FN + TP} \tag{4}$$

4.3 What is machine learning bias?

4.3.1 Bias and fairness

In this thesis, the term *bias* refers to cases where a machine learning model (a classifier) systematically makes disparate predictions between certain individuals or groups of individuals in terms of classification metrics [14]. In the literature, the terms bias and fairness are used frequently, both together and separately, to describe the same issues in machine learning [15]. In this thesis we will primarily use the term bias. However, some of the papers in the literature review use the word *fairness* to describe this concept.

4.3.2 Inductive bias

To avoid confusion, we briefly state that when we use the word bias, we do not refer to *inductive bias*.

Inductive bias is a design parameter associated with model performance where not enough bias leads to over-fitting, and too much bias leads to under-fitting. Inductive bias is necessary in all machine learning techniques. When building a machine learning model, there is a bias-variance trade-off in order to make a well-performing predictor, which is able to generalize well [9].

5 Literature Review

This section will review literature from the research area on bias in machine learning. The purpose of this review is to find identification and mitigation techniques that can be used to answer the research questions. After reviewing each paper, we reflect on how the techniques or theories relate to the purpose of our thesis.

Some of the papers reviewed have an exclusive focus on either bias identification or mitigation, while others present techniques for both identification and mitigation. To structure the review, we will first present papers that focus mainly on identification and then present papers that focus mainly on mitigation. This structure serves a purpose of building a vocabulary for bias identification and mitigation in a meaningful way since bias identification, to a certain extent, is a necessary precursor of bias mitigation and because being able to identify bias plays a central role in knowing whether bias has been mitigated.

5.1 Identification of bias

The papers reviewed in the following sections present techniques for bias identification.

5.1.1 Fairness Through Awareness

In their 2011 research paper, Dwork et al. study fairness in classification. They describe the challenge of fairness as *"achieving utility in classification for some purpose, while at the same time preventing discrimination against protected population subgroups"* [16]. They assess fairness on two levels, individual fairness and group fairness.

Individual fairness
They capture individual fairness *"by the principle that two individuals who are similar with respect to a particular task should be classified similarly"* [16, p. 1]. They state that a mapping $M : V \rightarrow \Delta(A)$, which maps individuals to a probability distribution over outcomes (A), satisfies the Lipschitz property for every x, y (pair of individuals) $\in V$, if:

$$D(M(x), M(y)) \leq d(x, y) \tag{5}$$

This ensures that the distance between predicted labels of any two individuals is less than or equal to the distance between the features that were used for prediction where $D()$ and $d()$ are some measures of similarity.

If incorporated into a classifier by adding it as a constraint in the optimization, individual fairness can be achieved by minimizing some arbitrary loss subject to the individual fairness constraint. In this case, the machine learning model output is a mapping that has to satisfy the Lipschitz property.

Group fairness
The authors capture group fairness or statistical parity as *"the property that the demographics of those receiving positive (negative) classification are identical to the demographics of the population as a whole"* [16, p. 2]. In other words, that the group receiving positive classification resembles the group receiving negative classification to a high degree. A mapping $M : V \rightarrow \Delta(A)$, satisfies group fairness between distributions S and T up to bias ϵ if:

$$D(M(s), M(t)) \leq \epsilon \tag{6}$$

Where $s \in S$, $t \in T$ and D is the statistical distance. The lower ϵ is, the more the groups receiving positive (negative) classifications should resemble each other.

Insufficiency of individual fairness and group fairness
Dwork et al. illustrate the insufficiency of only imposing either individual fairness or group fairness using a hypothetical distribution between two groups T and S illustrated in figure 1:

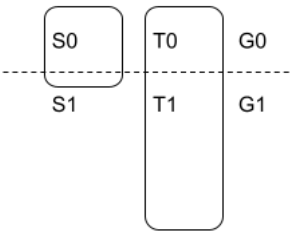


Figure 1: Hypothetical distribution of two groups [16]

In the figure, S and T denote two different groups; G denotes some treatment, for example, loan approval, while the dashed line is a threshold. Individuals in S0 and T0 are not approved for a loan (G0), while individuals in S1 and T1 are approved (G1). Dwork et al. show that imposing group fairness in this situation would either entail denying or approving all individuals in S and T (hereby treating groups in the same way) or denying/approving loans for S and T with the same probability. Dwork et al. also show that only imposing individual fairness could enable discrimination of S since the threshold could be set at a level so that no individuals in S are approved for a loan.

Furthermore, Dwork et al. highlight three ways in which group fairness is an inadequate notion of fairness:

- Reduced utility: By maintaining parity between two distributions, the utility of some classification can be reduced if there exists a desirable property that is very differently allocated between the two distributions.
- Self-fulfilling prophecy: By deliberately choosing an unqualified subset of a particular group, decision-makers can "justify" discrimination against this particular group in the future while maintaining statistical parity. This practice is used by some firms that have audited their hiring processes to ensure enough interviews with minority candidates.
- Subset targeting: Group fairness for distributions does not imply group fairness for subsets of the same distributions. Decision-makers can target a subset within the two distributions that is not equally represented in each of the distributions, hereby targeting a specific demographic deliberately while maintaining group fairness. For example, exposing an advertisement equally to two groups, does not inhibit a target subset (those who are expected to click on the add) from being very unevenly distributed between the two groups.

Relevance to the thesis
The Dwork et al. paper is highly cited and represents an early attempt at creating a theoretical framework that can deal with algorithmic fairness. The mathematical notions of individual and group fairness are beneficial concepts used in many of the papers in this review. Furthermore, the authors show the dangers of solely relying on satisfying individual or group fairness and how discrimination can still exist in perfect accordance within both notions of fairness.

5.1.2 Learning Classification without Disparate Mistreatment

Zafar et al. [17] introduce three notions of unfairness. The first two, *disparate treatment* and *disparate impact*, are applicable in scenarios where there is no available or reliable ground truth. The third, *disparate mistreatment*, is applicable when ground truth is available and reliable.

Disparate treatment
Disparate treatment arises when a model creates different outputs for groups that are similar on all characteristics except a sensitive attribute. Regarding gender bias, disparate treatment could correspond to the notion that two similar individuals should not be treated differently only because they have different genders [17]. For avoiding disparate treatment in a binary classifier, the following must apply:

$$P(\hat{y}|\mathbf{x}, z) = P(\hat{y}, \mathbf{x}) \tag{7}$$

Where x are non-sensitive attributes, y is the output, and z is a sensitive attribute, for example, gender.

Disparate impact
Disparate impact arises when a model's output benefits a group of people with the same values of a sensitive attribute more frequently than other groups. Concerning gender, disparate impact is when the fraction of males and females that benefit from a system is different [17]. To avoid disparate impact in a binary classifier, the following must apply:

$$P(\hat{y} = 1|z = 0) = P(\hat{y} = 1, z = 1) \tag{8}$$

Where z is a binary sensitive attribute.

Disparate mistreatment
The authors state that a model suffers from disparate mistreatment concerning a given sensitive attribute if the misclassification rates differ for groups of people with different values of that sensitive attribute.

A binary classifier does not suffer from disparate mistreatment if the misclassification rates are the same across groups with different values of the sensitive feature z. The misclassification rate can be described in several ways. Zafar et al. investigate disparate mistreatment in terms of the overall misclassification rate (OMR), false-positive rate (FPR), and false-negative rate (FNR). To avoid disparate mistreatment in a binary classifier, the following must apply for each of the above terms, respectively:

Overall misclassification rate (OMR)

$$P(\hat{y} \neq y|z = 0) = P(\hat{y} \neq y, z = 1) \tag{9}$$

False positive rate (FPR)

$$P(\hat{y} = 1|z = 0, y = 0) = P(\hat{y} = 1|z = 1, y = 0) \tag{10}$$

False negative rate (FNR)

$$P(\hat{y} = 0|z = 0, y = 1) = P(\hat{y} = 0|z = 1, y = 1) \tag{11}$$

Relevance to the thesis
The paper by Zafar et al. provides a bias identification technique. It does so in two scenarios: 1) when there is no reliable ground truth and 2) when there is reliable ground truth. In our thesis we assume that the ground truth is reliable (registered falls). Therefore, disparate mistreatment could be a relevant bias identification technique.

5.1.3 The Blinder–Oaxaca decomposition for linear regression models

The Blinder-Oaxaca decomposition method divides an observed difference in an outcome between groups into distinct components when using linear regression models. This decomposition can be used to identify potential discrimination or bias. In [18], Blinder-Oaxaca decomposition is used to study the differences in wages between men and women. The author, Jann, decomposes the difference into three components, seen from the viewpoint of women. To explain the decomposition method, we use the wage differences between men and women as an example. The differences regarding mean wage between men and women, R, can be written as:

$$R = E + C + I \tag{12}$$

The first component, E, relates to the part of the differential in outcome that is due to group differences in the predictors. Jann calls this the "endowments effect". E describes how the wage of the average woman would change if she had the predictors of the average man:

$$E = \{E(X_m) - E(X_f)\}\beta_f \tag{13}$$

The second component, C, is the contribution of the differences in the coefficients. C describes how the wage of the average woman would change if she was "treated" in the same way as a man or, in other words, had the same coefficients [18]:

$$C = E(X_f)(\beta_m - \beta_f) \tag{14}$$

The third component, I, is an interaction term. It accounts for the difference in endowments and coefficients that exist concurrently between the two groups:

$$I = \{E(X_m) - E(X_f)\}(\beta_m - \beta_f) \tag{15}$$

The three components together compose the outcome difference, R. All the equations above take the viewpoint of females. The difference, R, could also have been applied to the viewpoint of males.

Relevance to the thesis
The method could be used for bias identification since the decomposition explains the mean difference of the outcome variable, which in the AIR case can be translated to the mean difference in predicted risk of falling between, for example, male and female citizens. The decomposition technique is primarily used in a linear regression setting and could thus be relevant if we chose to use a linear regression model on the AIR data.

5.2 Bias mitigation

When reviewing bias mitigation techniques, we use a distinction from Calmon et al. [19] who define three areas of interest that can be used when assessing techniques to mitigate bias, namely pre-processing, in-processing, and post-processing. This distinction provides a meaningful way to relate the techniques to one another.

Pre-processing

5.2.1 Learning Fair Representations

In their 2013 paper Zemel et al. [20] propose a learning algorithm that attempts to achieve both group fairness and individual fairness. They formulate an optimization problem with the goal of finding an alternative representation that encodes the data as well as possible while obfuscating information about membership of protected groups (e.g. race or gender). In a way that *"lose/s/ any information that can identify whether the person belongs to the protected subgroup, while retaining as much other information as possible"* [20, p. 2].

The original data, X, is represented by k vectors, $\mathbf{v}_k$, that have the same dimensionality as x (the number of features). The vector, $\mathbf{v}_k$, is called a prototype, and the set of k vectors is called Z.

The original outcome, Y, is also represented by k vectors, $\mathbf{w}_k$, with the same dimensionality as y (one-dimensional). The algorithm attempts to learn values of $\mathbf{v}_k$ and $\mathbf{w}_k$ that achieve three explicit goals, which are that:

1. the mapping from X to Z satisfies group fairness
2. the mapping to Z-space retains as much information from X (except for membership of protected groups)
3. the induced mapping from X to Y (through Z) is as close to the original mapping between X to Y as possible.

$\mathbf{v}_k$ and $\mathbf{w}_k$ are the only two parameters that need to be learned.

When calculating the new values of an observation, $\hat{\mathbf{x}}_n$, the learned $\mathbf{v}_k$ is multiplied with a probability mapping $M_{n,k}$:

$$\hat{\mathbf{x}}_n = \sum_{k=1}^{K} M_{n,k}\mathbf{v}_k \tag{16}$$

$M_{n,k}$ contains a value related to each vector in the set of $\mathbf{v}_k$, governing the mapping between the intermediate Z-space representation of $\mathbf{x}$ and the new $\hat{\mathbf{x}}$. Each row in $M_{n,k}$ sums to 1 and can be interpreted as a probability mapping of a given observation to the k vectors.

If k=2, a combination of v_1 and v_2 is the Z-space representation of the original data. When calculating $\hat{\mathbf{x}}_n$ for the first person in the data set, then n=1. Then the new data for the person is found by:

$$\hat{\mathbf{x}}_1 = \sum_{k=1}^{2} M_{1,k}\mathbf{v}_k = M_{1,1} * \mathbf{v}_1 + M_{1,2} * \mathbf{v}_2 \tag{17}$$

When calculating $\hat{\mathbf{y}}_n$, the learned $\mathbf{w}_k$ is multiplied with the same probability mapping $M_{n,k}$:

$$\hat{\mathbf{y}}_n = \sum_{k=1}^{K} M_{n,k}\mathbf{w}_k \tag{18}$$

In order to achieve group fairness, a probability mapping is created for the protected group, M_k^+, and a mapping for the non-protected group, M_k^-.

The setup yields a learning system that attempts to minimize the following loss function:

$$L = A_z \cdot L_z + A_x \cdot L_x + A_y \cdot L_y \tag{19}$$

A_z, A_x and A_y are hyper parameters governing trade-offs.

L_z is used to achieve group fairness by ensuring that the difference between the mapping of individuals in/not in protected groups to the set of prototypes (M_k^+, M_k^-) is as small as possible:

$$L_z = \sum_{k=1}^{K} |M_k^+ - M_k^-| \tag{20}$$

L_x is used to achieve that the mapping of x into $\hat{\mathbf{x}}_n$ through Z is a good description of x, quantified by a squared error measure.

$$L_x = \sum_{n=1}^{N} (\mathbf{x}_n - \hat{\mathbf{x}}_n)^2 \tag{21}$$

L_y requires that the predictions are as accurate as possible:

$$L_y = \sum_{n=1}^{N} -y_n \log \hat{y}_n - (1 - y_n) \log(1 - \hat{y}_n) \tag{22}$$

By finding an intermediate representation of the data, the approach can be used for any classification task by applying a classifier to the intermediate representation of the data instead of the original data. In this way, the approach can be used to turn any classification algorithm into a fair classifier without having access to the algorithm itself. This is in line with the philosophy of the authors, which revolves around achieving fairness through establishing a *"two-step system construction by two parties: an impartial party attempting to enforce fairness, and a vendor attempting to classify individuals"* [20, p. 4].

The authors compare their model to the performance of other models with respect to *accuracy*, *discrimination*, and *consistency*. *Accuracy* measures the accuracy of model predictions, and *discrimination* measures bias with respect to the classifications and the sensitive feature(s) in the data. *Consistency* compares the model's classification of a given data point to its k-nearest neighbors. The discrimination measure is a form of groups fairness, while the consistency measures is a form of individual fairness. Results from tests on three different data sets show that their technique removes discrimination (achieves group fairness), maintains accuracy, achieves individual fairness, and successfully obfuscates information about membership of protected groups.

Relevance to the thesis
Zemel et al. present a pre-processing mitigation technique through learning fair representations of the data on which any classifier can be trained. Mitigating bias through the technique presented here seems relevant in the AIR case.

5.2.2 Optimized Pre-Processing for Discrimination Prevention

Calmon et al. [19] presents an optimization technique for producing transformations that can mitigate bias (which they call discrimination). The goal is to determine a mapping $P(\hat{X}, \hat{Y}|X, Y, D)$, where Y is an outcome, X denotes non-protected variables and D denotes protected variables. The mapping has the following goals:

1. Discrimination control, which governs the disparity in predictions between groups.
2. Distortion control, that ensures similarity between the original data and the transformed data.
3. Utility preservation, which controls that a model learned from the transformed data set is not too different from the one learned from the original data set.

According to Calmon et al. discrimination is : *"(...) the prejudicial treatment of an individual based on membership of a legally protected group such as a race or gender"* [19, p. 1]. Direct discrimination occurs when protected variables are used explicitly in decision-making, whereas indirect discrimination occurs when protected variables are not used, but variables correlated with them lead to different outcomes

for different groups. Simply removing a protected variable is not enough, because it does not address indirect discrimination - and can in fact conceal it.

Discrimination control
Discrimination control serves the purpose of limiting dependence of the transformed outcome $\hat{Y}$ on the protected variable D. In other words, that the distribution of the transformed outcome for any value of D, $p(\hat{Y}|D)$, resembles the distribution of the outcome for all groups, P(Y). This is ensured by adding a constraint to the optimization, where the ratio between the two distributions cannot be larger than a certain value that is controlled by a parameter ϵ.

Distortion control
Distortion control ensures that an observation (x) and its corresponding outcome (y) for a given person in the original data (x,y) should have values in the transformed data set ($\hat{x}$,$\hat{y}$) that are as close to the original values as possible. This constraint restricts the mapping to avoid certain larger changes - for example, that a very low credit score is mapped to a very high credit score.

Utility preservation
Utility preservation ensures that the model from the transformed data set is not too different from the one learned on the original data. Utility preservation is achieved through the definition of the objective function in the optimization that learns the transformed data set. The objective function is defined by a dissimilarity between the original distribution of (X, Y) and the new learned distribution of ($\hat{X}$,$\hat{Y}$). Therefore, minimizing the objective function will preserve the utility of the data set.

Experimental results
The paper has applied their approach to both the COMPAS data set and the UCI Adult data set. The approach has successfully decreased the discrimination but with a decrease in accuracy.

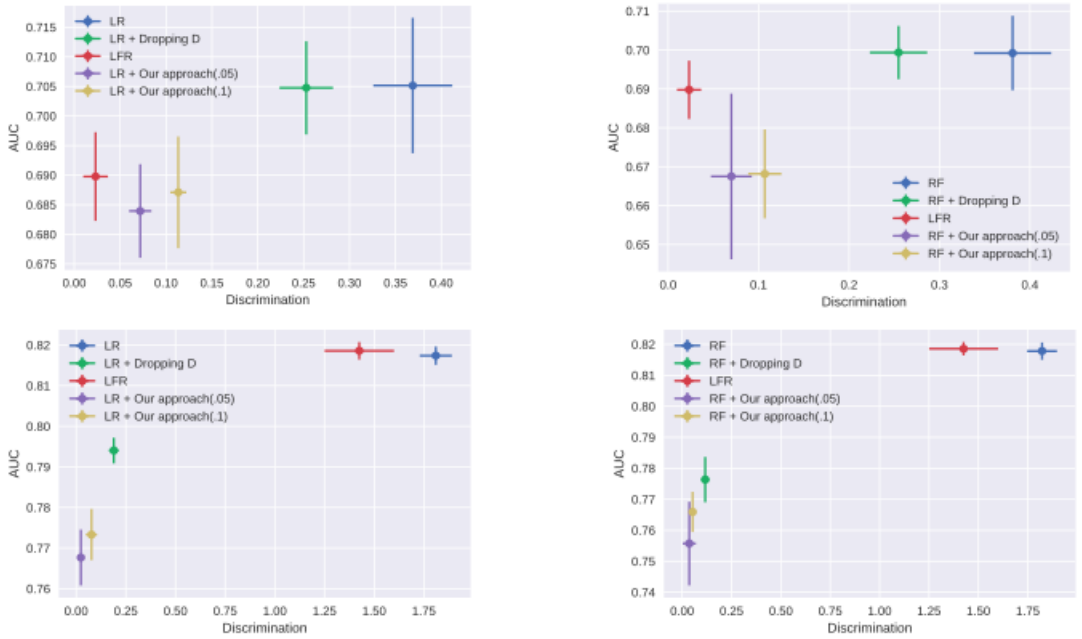


Figure 2: Top row relates results on the COMPAS data set and bottom row for UCI Adult data set. First column is logistic regression (LR), and second column is random forests (RF) [19].

The models without mitigation in both the LR and RF have the highest AUC but also the highest discrimination.

Dropping the protected variable (dropping D) reduces the discrimination of the models. In the COMPAS case, the AUC does not change, while it decreases the discrimination. In the UCI Adult case, the discrimination decreases almost to the same level as the approach of the authors, but with a higher AUC than Calmon et al.

Learning fair representation (LFR) (see section 5.2.1) reduces the discrimination the most on the COMPAS data set while preserving a higher AUC than Calmon et al.'s approach. However, on the UCI Adult data set, the effect of LFR is minimal.

Across the two cases, Calmon et al.'s approach for mitigating discrimination performs well in decreasing the discrimination but does so with a cost in AUC.

Relevance to the thesis
Calmon et al. present a technique to learn a transformation of the data that preserve utility and controls discrimination and distortion. They compare their approach to the approach of Zemel et al. (LFR) and to a simple mitigation technique of dropping the protected variable from the model. The techniques presented by Calmon et al. could be relevant for our thesis. Since Calmon et al. learns a transformed data set, which any classifier can use, hereby mitigating bias, the approach has much in common with the two-step strategy presented in Zemel et al.

5.2.3 Certifying and removing disparate impact

Feldman et al. [21] propose a method for identifying and removing bias. Their methods are rooted in the notion of the *80% rule*. The 80% rule is advocated by the US Equal Employment Opportunity Commission and states that a selection procedure violates the 80% rule if the group with lowest passing rate has a passing rate that is less than 80% of the passing rate of the group with the highest passing rate [22].

In a scenario where gender is the only relevant protected group characteristic, and we are assessing a hiring process, and men have the highest employment rate, then the 80% rule would be violated, if the employment rate of women is lower than 80% of the employment rate of men. This can be written mathematically as:

$$\frac{Pr(C = Hired|X = Female)}{Pr(C = Hired|X = Male)} \leq 0.8 \tag{23}$$

Where,

D = (X,Y,C), is a data set.

X: The protected attribute (for example sex, race, religion)

Y: The remaining attributes

C: Binary class (hired/not hired)

Feldman et al. adopt the same concept to be used on classification algorithms, in the sense that the predictions of an algorithm on groups with protected attributes must also comply with the 80% rule. If the algorithm does not, the authors state that it exhibits *disparate impact*.

Removing Disparate Impact
The paper by Feldman et al. [21] proposes a disparate impact removal algorithm that has the following goals:

- It should not be possible to predict the protected variable (X) from the remaining features (Y)
- Widely different outcomes for different groups related to a protected classes (disparate impact) should be avoided

The removal algorithm is run on the non-protected features Y and returns $\hat{Y}$. Combining $\hat{Y}$ with the original X and C gives an unbiased data set, $\hat{D}$=(X,$\hat{Y}$,C), also called a repaired version of D.
The repaired value of a non-protected variable, $\hat{Y}$, is found in the following way:

- Calculate the cumulative density function, F, and quantile function, F^{-1}, of a variable Y for every value of x ∈ X.
- Find the percentile score of an observation by putting Y into the F_x of the group x that the observation is in.
- Find the new repaired $\hat{Y}$ for the observation by taking the median of the results from putting the percentile score into the quantile functions for every x ∈ X. This "median" quantile function is called F_A^{-1}.

Repairing a non-protected variable in this way both preserves the within group ranking of observations, while it makes it difficult to separate the protected groups based on their values of the repaired non-protected variable.

Example: Finding repaired SAT scores with Disparate Impact Removal
We intend to find a repaired SAT score (entrance exam scores for universities and colleges), $\hat{Y}$, for females and males. We start by calculating the cumulative distribution functions and quantile functions for females and males that map SAT scores with percentiles and vice-versa:

$$F_{female}(SAT) = percentile \tag{24}$$

$$F_{male}(SAT) = percentile \tag{25}$$

$$F^{-1}_{female}(percentile) = SAT \tag{26}$$

$$F^{-1}_{male}(percentile) = SAT \tag{27}$$

A male who originally had a SAT score of 500 has an associated percentile of:

$$F_{male}(500) = 0.95 \tag{28}$$

His repaired SAT score is calculated as follows:

$$\hat{y} = median(F^{-1}_{female}(0.95),\ F^{-1}_{male}(0.95)) = 625 \tag{29}$$

Figure 3 shows the distributions of hypothetical SAT scores for females and males, where the black curve represents the repaired SAT score distribution.

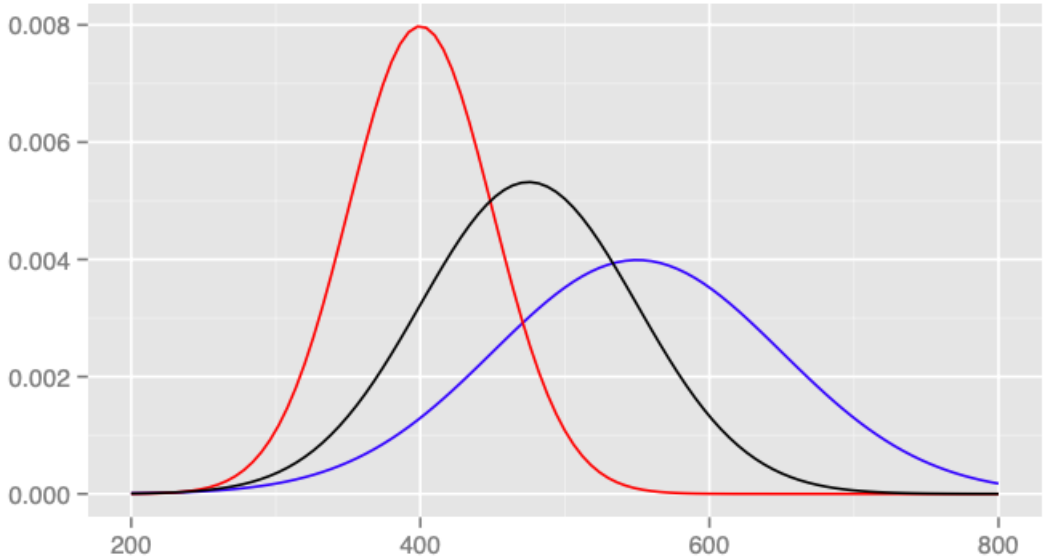


Figure 3: Hypothetical distributions of SAT scores for females (blue) and males (red). The fully repaired data is represented by the distribution in black [21].

Partial repair
Feldman et al. [21] also tested out partial repair techniques. A partial repair governs how much the distributions move towards each other. The repair ranges from 0 to 1, and the distributions are shifted towards the median distribution as the repair level increases. Repair level = 1 is the same as the repair method shown above.

Results
Feldman et al. [21] have tested their disparate impact removal technique on three data sets (Adult Income, German Credit, and Ricci data sets) that all have protected attributes in the data. Results show that across all data sets, using repair levels above 0.75 ensure that the tested classifiers comply with 80% rule.

Relevance to the thesis
Feldman et al. [21] propose a notion of fairness, disparate impact, that uses the relation between the positive predictions for two groups (for example, females and males). Furthermore, the authors use a threshold based on legally founded considerations about disparate impact, the so-called 80% rule. The notion *disparate impact* can be used for bias identification in the AIR data set, and the 80% rule could also be a tool for discussing the fairness of a data set. The authors' method for removing disparate impact can be used as a mitigation technique on the AIR data set. The use of disparate impact as a notion of fairness also relates to equal opportunity by Hardt et al. [23]. Hardt et al. use the notion when postprocessing a predictor, whereas Feldman et al. use the notion when preprocessing data for removing disparate impact.

5.2.4 Gender Bias in Contextualized Word Embeddings

Zhao et al. [3] examine the quantification and mitigation of gender bias in the field of NLP. In NLP, a model has learned vector representations of words. These vector representations, also called word embeddings, are the focal point of the Zhao et al. paper. They examine how two word embeddings, GloVe and ELMo, reproduce gender bias in their embedding values. They propose a mitigation technique, *gender swap*, that attempts to mitigate bias by data augmentation where gender related words are swapped. To evaluate whether mitigation has worked, they train an SVM on both the original and gender swapped versions of the data and assess its performance.

Experimental setup
The authors used the WinoBias data set (a word corpus), which is a data set consisting of sentences with entities corresponding to people, referred to by their occupation. For example, "the engineer went back to his home" is such a sentence, where the entity *engineer* is the occupation. WinoBias is created to explore gender bias and is divided into two subsets: pro-stereotype and anti-stereotype. Sentences in the pro-stereotype data set have pronouns associated with gender stereotypes (for example, "the nurse rides her bike"). In the anti-stereotype data set, the opposite is true (for example, "the nurse rides his bike").

To identify gender bias, the authors train an SVM to classify the gender of an entity in a sentence, based on the embedding values of the words used in the sentence. For example, "the nurse rides her bike" should be classified as "female". They test an SVM on both the pro-stereotypical and anti-stereotypical data sets. If the performance of the SVM tested on the pro-stereotypical data set is substantially better than the performance of the SVM tested on the anti-stereotypical data set, then the authors identify that gender bias has been propagated into the word embeddings.

To test whether gender swap mitigates bias, they train an SVM on the original data and on the gender swapped data set. If the performances of the SVM trained on the gender swapped data have similar performances on the pro-stereotypical and anti-stereotypical, then bias has been mitigated.

Gender swap
Gender swap is done by replacing gender-revealing entities in the data set with words indicating the opposite gender. For example, the sentence "the engineer went back to her home" has a swapped version: "the engineer went back to his home". The gender swapped version of the original data set is then concatenated with the original data, creating the gender swapped data set (illustrated in Fig. 4).

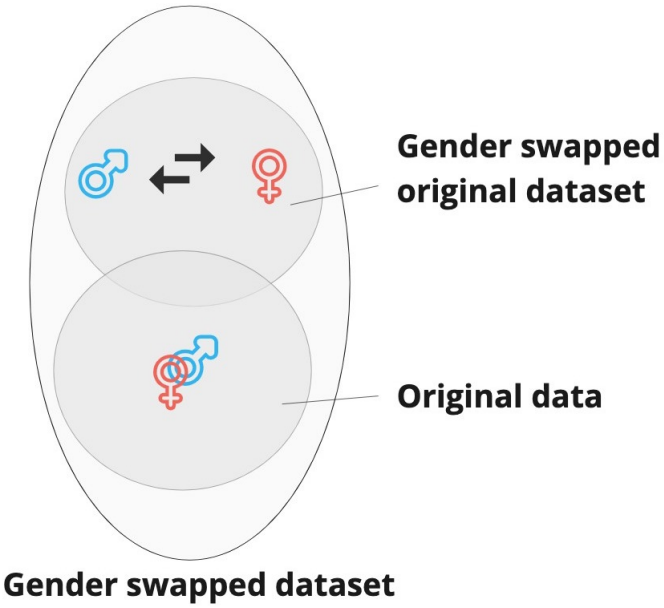


Figure 4: Gender swapped data set: The union of a gender-swapped version of the original data set and the original data [3]. Illustration made by Daniel Vigild Juhász and Lau Johansson.

Results
Table 2 shows the F1-scores of the SVMs trained on respectively the original data and the gender swapped data, when tested on the pro-stereotypical and anti-stereotypical WinoBias sentences.

Data	**Pro. (F1)**	**Anti. (F1)**	**|Difference|**
Original data set	79.1	49.5	29.6
Gender swapped data set	65.9	64.9	1.0

Table 2: The F1-score of SVM predictions on the pro-stereotype and anti-stereotype data.

Data augmentation reduces the difference in F1-score percentage points between the two data sets (pro- and anti-stereotype) from 29.6 to 1.0. The results show that gender swap is largely effective at mitigating the bias found in the embeddings of GloVe and ELMo [3].

Relevance to the thesis
The mitigation technique performed by Zhao et al. shows that swapping genders in the word corpus as data augmentation could reduce gender bias. Since the methods tested in the paper relate to NLP, and the bias is expressed in terms of how well the SVM performs in classifying the gender of an entity, the approach cannot directly be applied in the thesis. However, the data augmentation technique (swapping genders) could be tested on the AIR data set to examine if bias can be mitigated.

In-processing

5.2.5 Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings

In their 2016 paper, Bolukbasi et al. [24] point to the risk of amplifying biases present in word embeddings and offer a solution to debiasing word embeddings. They find evidence of gender biases in word embeddings. For example, MAN is to COMPUTER-PROGRAMMER as WOMAN is to HOMEMAKER. In their paper, they distinguish between two types of bias: *direct bias* and *indirect bias*:

- Direct bias: The authors describe direct bias as *"an association between a gender-neutral word and a clear gender pair"* [24, p. 3]. Here, they make an important distinction between *gender-neutral words* (for example, nurse, doctor, football, softball) and *gender-specific words* (for example, man, woman, uncle, aunt). Gender-specific words form gender pairs, for example, he-she and king-queen. An example of direct bias is that they find that the word embedding of *nurse* is much closer to *woman* than to *man*. In other words, there is direct gender bias when a gender-neutral word, e.g., *nurse*, has an unequal distance to a clear gender pair.
- Indirect bias: The authors describe indirect bias as *"associations between gender neutral words that are clearly arising from gender"* [24, p. 4]. An example of this is that the embedding of the word *receptionist* is closer to *softball* than to *football*. Here, *receptionist* and *softball* are typically associated with females, while *football* is associated with males.

To illustrate the potential harms of these biases, they offer a hypothetical example of an algorithm with the task of retrieving relevant web pages for some query. If word embeddings are used to improve the order of web page results, then the biased embeddings could have real-world outcomes. For example, when searching for qualified candidates for a job opening as a computer programmer, the embeddings will lead to gender-biased search results that disadvantage female candidates.

Debiasing
The authors present debiasing as a way to reduce gender bias in the word embeddings. The goals of debiasing are as follow:

- Reduce bias: a) *"ensure that gender-neutral words such as nurse are equidistant between gender pairs such as she-he"* [24, p. 4] and b) reduce biased gender associations between gender-neutral words.
- Maintain embedding utility: a) maintenance of meaningful associations that are not gender related between gender-neutral words, and b) to maintain definitional gender associations, for example, between *man* and *father*.

The first step towards debiasing is identificaiton of the *gender subspace*, which is a direction in the embedding space that describes the diffuse female-male notion. By aggregating over multiple differences in word pair directions, for example, $\overrightarrow{she} - \overrightarrow{he}$ and $\overrightarrow{woman} - \overrightarrow{man}$, and computing their principle components, the authors identify a single direction, which explains the majority of the variance.

This direction g is defined as a general gender direction that can be used to calculate both direct and indirect bias and to debias word embeddings. Here, the authors present two ways of debiasing: hard debiasing and soft debiasing.

- Hard debiasing: Ensures that all gender-neutral words have value zero in the gender direction g and that the words are equidistant to all words in the gender specific word sets, for example, man-woman, grandmother-grandfather etc.
- Soft debiasing: Neutralizes embeddings and equalizes to a degree, using a trade-off parameter, while maintaining as much similarity to the original embedding as possible.

When evaluating the performance of hard debiased and soft debiased embeddings for different tasks, the authors find that the new embeddings and transformations do not negatively impact the performance. They test the embeddings by assessing the characteristics of their generated analogies.

In figure 5, the original embeddings generate the most stereotypical analogies (blue), and by applying hard debiasing (green), the analogies become less stereotypical. Consider the analogy puzzle, HE to DOCTOR is as SHE to X. The original embedding returns X = NURSE, while the hard-debiased embedding finds X = PHYSICIAN.

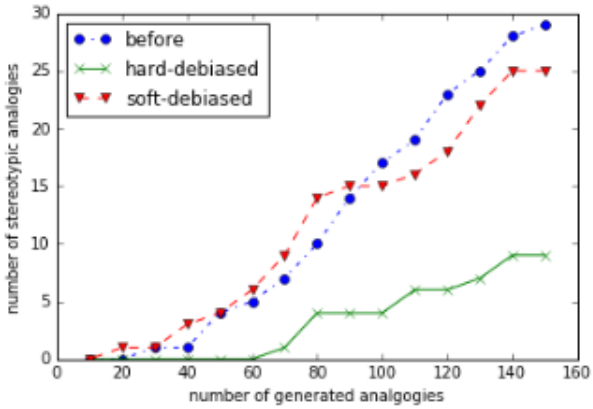


Figure 5: Debiasing word embeddings [24].

Relevance to the thesis
The paper by Bolukbasi et al. [24] relates to bias mitigation in the field of NLP. The natural structure of language has (in some cases) incorporated gender associations (for example *father* and *queen*). The modeling task of the AIR case is not NLP related, and therefore, the technique cannot be applied to the AIR model. However, the review of this article gives insight into how bias is mitigated within other machine learning areas.

Post-processing

5.2.6 Equality of Opportunity in Supervised Learning

Hardt et al. [23] propose two ways of bias identification with respect to a protected attribute and a method for post-processing bias mitigation by learning a fair predictor from the predictions of a model.

Hardt et al. introduce two notions related to bias: *equalized odds* and *equal opportunity*.

Equalized odds
"A predictor $\hat{Y}$ satisfies equalized odds with respect to protected attribute, A, and outcome, Y, if $\hat{Y}$ and A are independent conditional on Y." [23, p. 2]. The paper focuses on binary targets, and in that case, equalized odds is equivalent to:

$$Pr(\hat{Y} = 1|A = 0, Y = y) = Pr(\hat{Y} = 1|A = 1, Y = y), y \in \{0, 1\} \tag{30}$$

Equalized odds requires that a classifier generates equal true positive rates and equal false positive rates across two demographics (A=0 and A=1), e.g., females and males. According to the authors, if a predictor satisfies equalized odds, then it would be deemed "unbiased".

Equal opportunity
Equal opportunity is a relaxation of equalized odds since it only requires equal true positive rate between two demographics:

$$Pr(\hat{Y} = 1|A = 0, Y = 1) = Pr(\hat{Y} = 1|A = 1, Y = 1) \tag{31}$$

The notion comes from the emphasis of Y=1 being the advantageous outcome, for example, receiving a loan or promotion. The notion then requires that both groups have an equal opportunity for the advantageous outcome. The relaxation is weaker and therefore less "unbiased". However, it allows for better utility.

Deriving the unbiased predictor
The authors mitigate bias as a post-processing step, which means that the data and the model is left unchanged. They construct an unbiased predictor, $\tilde{Y}$, by deriving it from $\hat{Y}$.

The unbiased predictor can be derived by a linear program where the objective is to optimize the accuracy between $\tilde{Y}$ and Y. The program has two constraints: 1) the TPR for females and males should be equal, and 2) the FPR for females and males should be equal. The two constraints together form the equalized odds constraint in equation 30.

The unbiased predictor can be derived from a model's probability output $\hat{Y}$. The goal is to find an optimal threshold that satisfies the equalized odds constraint. The optimal solution can find different thresholds, t_a, one for each different value of A (e.g. females and males). However, if the ROC curves do not intersect, then the optimal solution may require randomizing between two thresholds for each group. The unbiased predictor can also be derived by only using the equal opportunity constraint.

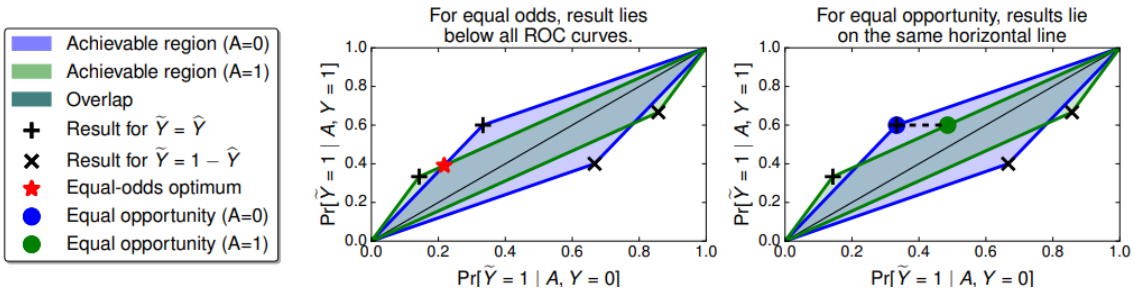


Figure 6: Left: Finding optimal equalized odds predictor. Right: Finding optimal equalized opportunity predictor. First axis is false positives rates and second axis true positives rates. [23]

Figure 6 shows the trade-off between false positives and true positives for derived predictor $\tilde{Y}$. The blue area is an achievable region for group A=0 and the green for A=1. If equalized odds is required for a predictor, then a solution to the linear program is exactly at the intersection between the blue and the green region (the red star). Since equal opportunity is a relaxation, only the true positive rates should be equal. This is shown in figure 6, where the blue and green dot are horisontally aligned.

Results
Hardt et al. test the unbiased predictor on a case study of FICO scores. These scores are used for classification purposes in the US to predict creditworthiness. The authors train a classifier with no constraints and use the performance of this model as a benchmark. A classifier subject to the equal opportunity constraint achieves 92.8 % of the utility of the benchmark classifier, while a classifier subject to equal odds achieves 90.2 % [23].

Relevance to the thesis
Hardt et al. present two measures for identifying bias: equalized odds and equal opportunity. Equalized odds is equivalent to the notion of disparate impact from the Zafar et al. paper in section 5.1.2 although Hardt et al. assume a ground truth (y) to be known. In their notion of equal opportunity, they relax the equalized odds expression, so it only has to be valid for observations where the ground truth is $y = 1$. The techniques presented by Hardt et al. could be relevant in the AIR project, since once can assume that the ground truth $y = 1$ is known.

6 Case description: The AIR project

This section describes the purpose of the AIR project, the organisational and political context and presents a flow diagram that explicitly shows where and how the AI (AIR classification algorithm) is expected to support the caseworkers in their decision-making.

The AIR project (AI Rehabilitation) is a project focusing on using AI for decision support during the rehabilitation process of elderly citizens in Aalborg Municipality [25]. It is one of several signature projects that are initiated by the Danish government, Local Government Denmark and Danish Regions. The signature projects will test the usage of artificial intelligence in the Danish public sector [26]. The project is in a development phase and therefore not implemented as of 2021.

6.1 Purpose

The AIR project seeks to create a *"solution based on models and statistics that can support the individual caseworker's assessment of municipality services regarding rehabilitation, particularly training, better use of aids and fall prevention training"* [25]. We focus on a part of the AIR project that explores developing an AI that can help caseworkers answer the following:

- Who should be offered fall prevention training?

A key component in answering the question above is the data regarding the granted aids of each citizen. Aalborg Municipality registers all aids granted to each citizen, and these form, in the words of the project owner, an *"aid-DNA that tells a story about the impairments of the individual citizen"* [2, translated from Danish]. It is this *aid-DNA* (combinations of aids provided) coupled with data regarding the citizen that are expected to contain patterns which the AI can use to make predictions that can help the case workers determine if the citizen should be provided with fall prevention training [2].

6.2 Organisational context

The AIR project is placed within the Referral Unit - Support and Care of Aalborg Municipality, with Aarhus University and DigiRehab as collaborators. Aarhus University is responsible for developing the AI, and DigiRehab is a private company that Aalborg Municipality has hired to provide a digital platform which their physiotherapists and homecare workers can use. The Referral Unit deals with the care of citizens who have *"long lasting and continuous needs for rehabilitation, enabling them to - in a strengthened way - live a partially independent and meaningful everyday life with assistance from compensated services and aids"* [27, translated from danish]. The unit has two departments, each with 20 employees: one department deals with referring citizens to rehabilitation, and the other department assesses what type of service or aid will help the citizen [2]. As of 2021, the unit manages the care of 2.600 citizens.

A key outcome of the AIR project is that the AI will, hopefully, help the caseworkers by identifying citizens in an automated fashion that are likely to benefit from services. Since manually handling each case is very laborious, algorithmic monitoring of the citizens under the care of the Referral Unit could lead to even more effective identification of the citizens' needs [2]. For example, in the case of predicting the probability of falling, using the AI could help caseworkers identify an *aid-DNA* that correlates with a high risk of falling and offer fall prevention training before the falls occurs. In this light, successful development and implementation of the AIR project would lead to a significant leap both in terms of efficiency and effectiveness for the Referral Unit. The AI can be used to support identifying which citizens to contact and save time in terms of monitoring the 2.600 citizens they have under their care [2].

6.3 Political context

In the Consolidation Act on Social Services of 2015 by The Danish Ministry of Social Affairs and the Interior, article 83a states that the *"municipal council shall offer a brief and time-limited rehabilitation program to individuals with functional impairment, if the rehabilitation program is assessed to be able to improve their functional impairment and thus reduce the need for assistance"* [28]. The project owner confirms that the AIR project can be viewed as a part of Aalborg Municipality's effort towards complying

with article 83a [2]. The reasoning behind offering brief and time-limited rehabilitation is two-fold: 1) it can improve the individual's functional impairment and 2) reduce the need for future assistance. In this light, the AI will guide the Referral Unit in their search for citizens who are expected to benefit from a brief and time-limited rehabilitation program. From the article text and interview with the project owner, it is clear that the political intent of AIR project is to deliver more welfare to citizens, not to optimize within the current budget.

Moreover, an AIR project team member states that they are more cautious with mistakes regarding not providing fall prevention training when it is needed than mistakes regarding providing training to citizens who might not need it [29]. In terms of classification rates, they are more willing to accept higher false positive rates (FPR) than higher false negative rates (FNR); the consequences of a false positive and false negative classification support this idea. While a false positive classification would result in the potential waste of a fall prevention training program, a false negative classification would mean withholding valuable training to a citizen who has a high risk of falling. Since falling can have very severe consequences for elderly citizens, it is clear why the AIR project team chooses to value misclassifications in the way described here.

Finally, citizens have the rights to access records and documents that describe a decision or case regarding the citizen [30]. Therefore, the municipality could be requested by citizens to provide information regarding bias identification and mitigation efforts. In this light, simple and intuitive methods might be easier to communicate to citizens and therefore preferred over more complex methods.

6.4 Flow diagram of the AIR project

This section describes the process of a citizen entering the Referral Unit's area of authority related to the use of the AI model. A citizen can come from multiple sources in the welfare system, for example, after an accident or functional impairment due to aging. The AI will typically screen citizens who have been given an aid, although, this is not strictly mandatory in order to be screened and provided fall prevention training. Figure 7 shows how the typical process of screening and predicting the risk of falling is envisioned.

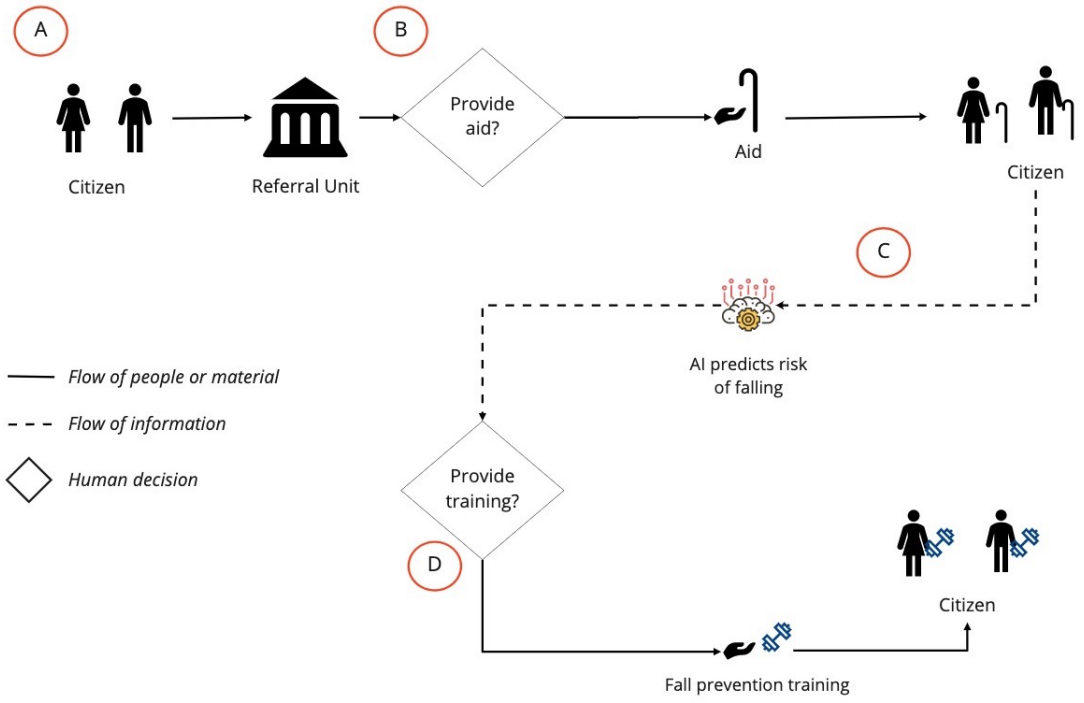


Figure 7: A process flow diagram of the AIR project related to assessing whether citizens should be provided fall prevention training.

A: The Referral Unit receives a case concerning a citizen. Typically, the citizen applies for one or more aids that the Municipality can provide.

B: If needed, the Referral Unit decides which aid(s) the citizen should be provided e.g., a walking stick or a wheelchair.

C: The provision of the aid is registered, and the data is sent to the AI, including information about e.g., the age of the citizen. After receiving the information, the AI predicts a risk score which is used by the Referral Unit to assess, whether this *aid-DNA* represents a well-known pattern, implying that the citizen has a high risk of falling. This risk score is expected to be based on the probability of falling.

D: The output of the AI (risk score) is used as a tool for decision support when assessing whether a citizen should be provided with fall prevention training. The AI does not make any automated decisions. The risk score is included as an element in an overall assessment of the individual citizen, where other factors than the risk score are considered. Each citizen is assessed individually. If it is decided that a citizen should be provided fall prevention training, the training is ordered from either another municipal service or DigiRehab. The training program typically lasts for three months, where data regarding training and potential fall incidents are registered on the digital platform provided by DigiRehab.

6.5 The AIR data set

This section describes the data generating process and the content of the AIR data set. All information regarding the data set comes from a yet to be published internal document describing the AIR project and meetings held with Aarhus University [31][29].

6.5.1 Data collection process

The AIR classification model is trained on a data set containing 2144 records with information regarding citizens, their aids, and whether they have fallen. All data points are collected at the screening time, except whether or not the citizens have fallen, which refers to whether a fall has occurred within three months after the screening date. The screening is done by DigiRehab. Some of the observations in the data set are screenings of citizens who have been in other municipally subsidized programs before or have been in a DigiRehab program before. They could therefore be in the data set more than once. The data are collected during 2019 and 2020.

6.5.2 Features of the AIR data set

The features in the AIR data set are described in the table below:

Variable	Description	Type
Gender	The gender of the citizen (0: female, 1: male).	Binary
Age	Age of the citizen.	Integer
Cluster0-19	Result of k-modes clustering of aids into 20 clusters. Cluster information is one-hot-encoded (e.g. Cluster15 = 1 if observation is in cluster 15).	Binary
LoanPeriod	Average number of days that the citizen has borrowed aids from Aalborg Municipality.	Integer
NumberAts	Number of aids that the citizen has borrowed from Aalborg Municipality.	Integer
Ats_*	One-hot-encoded features for each of the 114 unique aids in the data set (e.g. if a citizen has a walking stick then Ats_walking_stick = 1).	Binary
Fall	Whether the citizen has fallen within three months after the DigiRehab screening. (0: "No fall", 1: "Fall")	Binary

Table 3: Description of AIR data set

Most of the features are self-explanatory, but a more detailed description of the features Cluster0-19 is provided in the following.

6.5.3 The aid cluster values

The aid cluster values are calculated by clustering the first 50 aids of each citizen into 20 clusters. The clusters are calculated using K-modes since the aid information is categorical data. K-modes uses a distance measure, for example, Hamming distance, to calculate the differences between the cluster centers and a given observation. These distances are used when clustering the observations. An important note is that not all citizens in the data set have 50 aids, but the algorithm uses the 50 first aids of each citizen, at the most. The sequence of the aids is also taken into account by the clustering algorithm. The clusters are finally one-hot-encoded.

6.5.4 Protected features in the AIR data set

From table 3 we identify two potential protected features: gender and age. Both are frequently used in the literature as examples of protected characteristics. However, **we choose to move forward with gender as the protected variable** since it has a more obvious group distinction (females/males). Finally, consideration regarding the scope and time limits of the thesis constrains us to choose only one protected characteristic for further analysis.

6.6 The AIR classification model

Aarhus University has developed the AIR classification model, which is intended to be used to predict the risk of falling, as shown in figure 7. They have chosen to build the AI using an XGBoost model. See section 8.3 for an explanation of the XGBoost model.

7 Method

The thesis examines how bias can be identified and mitigated in models used for decision support in the public sector, with the AIR project as the primary subject of analysis. This section describes our method for answering the research questions (see section 3). Followed by a description of our process regarding the literature review, and a presentation of how we obtained the data used in the AIR project.

7.1 Quantitative method

Mathematical modeling is a method where mathematics is applied to elucidate problems in the real world. Furthermore, the method consists of analysing models as well as developing mathematical models [32]. Mathematical modeling can help answering the research questions since building machine learning models on the AIR data set and analysing the results can elucidate how bias can be identified and mitigated in the AIR project.

Throughout the thesis, the AIR data set is used to build machine learning models and make descriptive analysis for understanding the data related to the AIR project's algorithm. We have not been a part of the data collection process but received a pre-processed AIR data set ready for analysis. The data is owned by Aalborg Municipality and is, therefore, public register data, which can be considered a reliable data source [33]. Quantitative method requires that results are measurable (quantifiable) [33]. Using quantified register data (AIR data set) and building machine learning models on the data provides quantifiable results, which is why we use quantitative method.

7.2 Interview and online meetings

We conducted a semi-structured interview [34] with AIR project owner Camilla Fibiger Smed (the respondent), from Aalborg Municipality, to obtain general information about the AIR project. Prior to the interview, we prepared some questions that we anticipated would help getting a greater understanding of the AIR project. Since our knowledge regarding the AIR project was limited, we let the respondent answer the questions openly. Furthermore, we were able to ask in-depth questions as we gained more insight into the AIR project. The semi-structured interview is a qualitative method [35] and was used in the thesis in order to gain knowledge about the AIR project that otherwise would not be available. The interview is referenced as [2]. Christian Marius Lillelund is a member of the AIR project and is one of Aarhus University's employees responsible for building the AIR machine learning model. We have been in close contact with Christian through several online meetings. These have not been recorded, but we refer to [29], when information is obtaining through the meetings. We have primarily used Christian to answer questions regarding the data generation process and the AIR machine learning model. In the thesis, Christian is often mentioned as the AIR project member from Aarhus University. Christian prepared a document with a summary of the information regarding the AIR data set that he has relayed through our meetings. The document is called *The internal AIR report* and is referenced as [31]. The report can be found in appendix, section M.

7.3 Obtaining data

7.3.1 Overall process

The master thesis project period runs between ultimo January 2021 and ultimo June 2021. In December 2020, the first virtual meeting with Aarhus University was held.

Our expectation was that we could obtain the data used in the AIR project in the early stages of the project period, but due to circumstances regarding data protection and permissions outside of our control, the data transfer happened the 10th of May.

The data set contains personal information about citizens in Aalborg Municipality (pseudonymized). Initially, a data processing agreement between Aalborg Municipality and DTU was made, where DTU Compute's Head of Department had to sign. Then, we had to enter into a contract with DTU Compute. Furthermore, DTU Compute had to create a server where the data could be stored in a secure manner. To access the data, a virtual machine was created by DTU, which we could access through SSH (secure

socket shell) from our own computers.

While waiting for the data, we had to find a way to continue the project without the AIR data in a fruitful manner, so that the master thesis would not be delayed. We therefore chose to find a placeholder data set on which we could test identification and mitigation of bias. Furthermore, we also reviewed literature related to qualitative theory regarding bias in algorithms. This part of the literature review was however not included as a part of the final literature review presented in the thesis but can be found in the appendix, section G.

As the placeholder data set, we chose to use the COMPAS data set, related to the COMPAS algorithm used for decision support in pretrial release rulings in the US. The COMPAS case is widely referenced in the literature as a case of bias in algorithmic decision support. The COMPAS data set contains approximately 7000 records of defendants who have committed a crime. The defendants have been assigned a COMPAS score; a score which represents the risk of the defendant re-offending. When used in practice, a judge will use the predicted score as a decision support tool in her/his ruling regarding pretrial release of a defendant [36]. Our analysis of the COMPAS data can be found in appendices H, I and J.

7.3.2 Detailed timeline of obtaining the AIR data

December 2020 - before the official beginning
At the beginning of December 2020, before the official start of the master thesis period, we held a meeting with two members of the AIR steering group who work at Aarhus University. They have developed the machine learning model which is intended to be used in the AIR project. They anticipated that we could get the data as a CSV file since the data was anonymized. However, we needed to get acceptance from the rest of the steering group. We were therefore invited to a meeting with the rest of the steering group, which was to be held in February.

February 2021
The steering group decided to postpone the steering group meeting with us since a group member could not attend. The meeting was postponed to 9th of March. Since the meeting was postponed, we chose to review literature related to bias in machine learning. Furthermore, we chose to reach out to the AIR project owner in order to get information about the project. The interview with the project owner was conducted on the 8th of February.

March 2021
On the 9th of March, the steering group meeting was held, where we presented what we anticipated to examine. The steering group found our thesis relevant for the AIR project. DigiRehab contacted a lawyer from Aalborg Munipacility who was expected to form a contract with us. We wrote an email to the lawyer, who replied that the data set contained personal information [*personhenførbar* in Danish]. We had to describe our project, what data we required, and how we intended to use it. Furthermore, the lawyer wrote that the contract would formally be agreed upon between Aalborg Municipality and DTU - not us as students. A general description of the data was then added to the contract. On the 19th of March, the contract was sent to DTU's legal department for proofreading.

April 2021
On the 14th of April, DTU's lawyer evaluated the contract descriptions. They had some legal questions to the lawyer from Aalborg Municipality. Especially, the lawyers from both DTU and Aalborg Municipality agreed that the data did indeed contain personal information. Therefore, a secure server had to be created to enable us to work securely with the data. Furthermore, the DTU lawyers wrote that the head of DTU Compute should sign the contract. This was done on the 26th of April. Ten days before, on the 16th of April, we had made a contract with DTU Compute to gain access to the data when it was transferred to the secure server.

May 2021
On the 3rd of May, we signed an agreement to get access to the secure server with an associated virtual machine. The same day, the server and virtual machine were running. On the 10th of May, we finally received the actual AIR data set. On the 12th of May, data was downloaded on the secure server. We contacted an AIR project member from Aarhus University that had been responsible for transforming

the data. On the 16th of May, with the help of the project member, we managed to clone the AIR project GitLab repository to the secure server. We finally had the fully transformed AIR data in our possession.

8 Theory

The following sections describes our bias identification approach, the mitigation techniques, and the models we use to create estimates of the classification rates on the AIR data set.

8.1 Bias identification

In the following, we describe how classification metrics are used for bias identification, and how the relation between classification rates grouped by genders can be exploited for identifying gender bias. The comparisons between classification rates of genders will be used to answer RQ1.

Bias classification metrics
In the literature review related to bias identification, we found two overall ways of defining bias. The approaches depend on whether the ground truth is available and reliable. If the ground truth is not available and reliable, then bias must be identified using only the predicted outcome. If the ground truth is available and reliable, bias can be identified using both the ground truth and predicted outcome. Since the actual falls of the citizens in the AIR data set are known, it is possible to examine the predicted falls of the classification models and compare them to the actual falls. For identifying bias the papers in the literature review use combinations of classification metrics of the definitions from table 1 (TP, FP, TN, and FN) [17] [21] [23]. When identifying bias, the papers in the literature review compare the actual and predicted outcomes between demographics, sometimes called protected groups [11], for example, females and males. We choose to focus on the both correct classifications (TP, TN) and misclassifications (FP, FN). Furthermore, since the analysis focus on gender bias, the classification metrics are compared between females and males. To ensure the gender specific metrics are comparable, we use the classification rates: TPR, TNR, FPR, and FNR. This approach follows [17] [23], who also use classification rates for identifying bias.

Bias identification technique
The bias identification approach of the thesis consists of two steps. First, we estimate the classification rates (TPR, FPR, TNR, and FNR) grouped by females and males, respectively. Then, **the confidence intervals of the estimates of classification rates are compared between the genders** to identify if there is a difference. If the confidence intervals do not overlap, we deem it sufficiently probable, that the difference between the genders are large enough to merit further identification of bias.

Second, the relations of the metrics between the genders are assessed using the 80% rule from Feldman et al. [21] who use the rule to identify bias regarding *disparate impact*. In the thesis, we do not assess disparate impact since we assume that the ground truth is reliable, and if males fall at a higher rate, a classifier where bias has been mitigated should still reflect this fact. If men fall at a higher rate than women, we do not consider this as bias. In stead **we choose to use the 80% rule on the classification metrics: TPR, TNR, FPR, and FNR** because the 80% rule provides a region where the relation between the estimates could be problematic in relation to bias. To calculate the relation of a classification rate, the estimate of the rate for females is divided by the estimate of the rate for males. The relation should comply with the 80% rule, which gives a lower boundary of 0.8 and upper boundary of 1.25. These boundaries should be seen as "guidelines" and not a strict threshold when assessing bias. The region between the boundaries is called the *80% region*. In this light, if the relation between the genders' classification rates are clearly outside the 80% region, a classifier could be biased in terms of the specific metric.

8.2 Bias mitigation

To answer RQ2, we have chosen four mitigation techniques from the literature review that we intend to test on the AIR data set. These four are: **dropping the gender variable, gender swapping, disparate impact removal, and learning fair representations**. After using a mitigation technique on the AIR data set, we will train a model on the resulting data set and test the model on the original data. The classification rates of the model built on the bias mitigated data set will be compared with the original classification rates. The mitigation techniques are chosen on the basis of two considerations:

varying the level of complexity and relevance for the AIR project and the future work on bias mitigation.

We have deliberately chosen techniques on different levels of complexity to test if adding complexity will lead to better results. In the AIR context, simple and explainable methods are preferred since it eases the strain for non-technical employees in the municipality who are expected to work on the project, and makes it easier to communicate the processes behind a given decision regarding a citizen in Aalborg. When making recommendations to the AIR project team regarding identifying and mitigating bias, both the level of complexity and the effect of the mitigation techniques will be taken into account.

Furthermore, we have chosen only to test pre-processing mitigation techniques. The Zemel et al. paper [20] presents a philosophy of a two-step system construction where one party attempts to mitigate bias and another party attempts to maximize their utility of classifications on the data set. Practically, this would imply that one party performs some mitigation efforts on a data set, and then sends the resulting data set to another party that learns a model on the data. We are inspired by this system since it makes the incentives clear for both parties, where one party only wishes to mitigate bias and another party solely focuses on the specific classification task at hand. In this way, the potential trade-off between bias mitigation and utility of the classification is no longer placed on a single party. This approach will also work if Aalborg Municipality wishes to collaborate with private organizations and companies, where the algorithms used for prediction might be proprietary. If a two-step system is desired, then only pre-processing techniques are applicable, since the mitigation efforts can only be made on the data set before it is sent to the other party and fed into a given model.

Dropping the gender variable

The idea behind dropping the gender variable comes from the Calmon et al. paper [19], where they use dropping gender as a benchmark model on which to compare their own mitigation approach and Zemel et al.'s learning fair representation technique. The authors note that dropping the protected variable might not always be an effective strategy since other variables that are not dropped could be highly correlated with the protected attribute, which would still enable bias.

We find the simple nature of the technique compelling, which also has merit in the context of the AIR project, where explainability is important. However, we acknowledge the potential issues with the technique, particularly regarding other attributes highly correlated with gender. Despite these issues, we find it relevant to test the technique on the AIR data set. Figure 8 shows the process of implementing the dropping gender technique and training/testing the models.

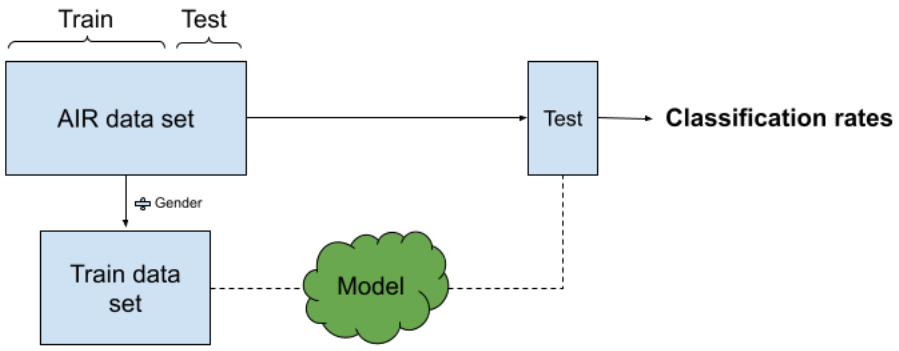


Figure 8: Dropping gender

As can be seen in the figure, we remove gender from the training set. Afterwards, a model is trained on the training data set without gender and tested on the test set.

Gender swap
The idea behind gender swap comes from the Zhao et al. paper [3], where they use the technique on sentences from a word corpus (WinoBias) that refer to females and males. In the AIR case, the data is in tabular form, and gender swapping the original data set is done by taking all rows for females and changing their gender to male and vice-versa. In the Zhao et al. paper, gender swap effectively mitigated the biased prediction of a classifier trained on the embeddings of GloVe and ELMo. The data structure and learning setup is different in the AIR case, but the technique is still relevant to test. Again, the mitigation technique is simple and intuitive, which is a compelling characteristic in the AIR context. Figure 9 shows the process of implementing gender swap and training/testing the models.

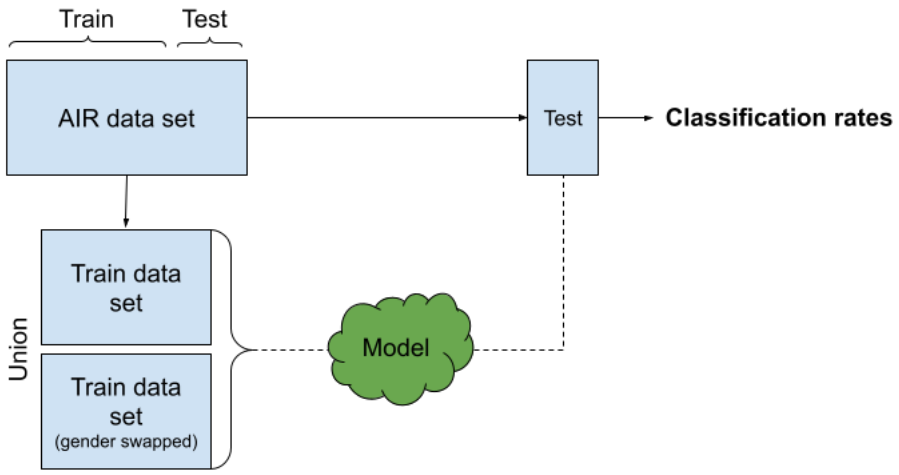


Figure 9: Gender swap

As the figure shows, we take the training set, copy it, swap the genders, and train the model on the union of the original training set and the gender swapped training set. The model is then tested on the test set.

Disparate impact removal
The idea behind disparate impact removal comes from the Feldman et al. paper [21], where they use the technique to ensure that is not possible to predict a protected variable from the remaining features. This is intended to have the effect that widely different outcomes for different groups related to a protected class is avoided. Feldman et al. operationalizes "widely different outcomes" by the 80% rule. It is important to note that our intended goal of the disparate impact removal technique is **not** to remove differences in the outcomes between genders since we do not consider it biased if there is a difference in the ground truth between females and males as described above. However, if the models built on the original AIR data set yield biased classification rates, then one could imagine that augmenting the data in a way that makes the distributions of variables between females and males as similar as possible might mitigate the biased classifications. Finally, we highlight that the disparate impact removal algorithm can only be used to change the numerical values, which in the AIR case is *Age*, *Loan period*, and *Number of aids*. The remaining features are one-hot-encoded and will therefore be left unchanged when the technique has been used on the data set. Figure 10 shows the process of implementing disparate impact removal and training/testing the models.

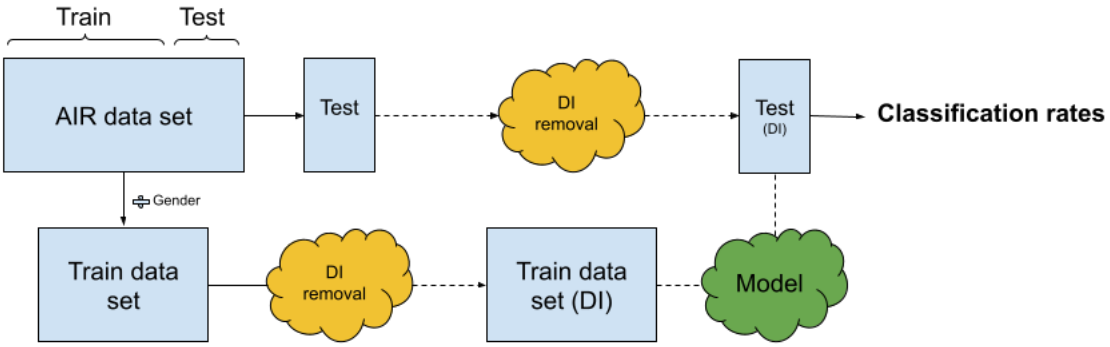


Figure 10: Disparate Impact Removal

As can be seen in the figure, we drop gender from the training set and run the disparate impact removal algorithm on the training set. Then we train a model on the resulting data set and test the model on a disparate impact removed test set.

We use an implementation of disparate impact removal from AIF360's pre-processing algorithm package [37].

Learning fair representations

The idea behind learning fair representations comes from the paper of Zemel et al. [20], where they find an alternative representation of the data that encodes the original data as well as possible while obfuscating information regarding the membership of protected groups. The purpose of the technique resembles the disparate impact removal algorithm, in the sense that it should not be possible to guess whether an observation is female or male based on the non-protected attributes. However, learning fair representations is a more complex model since it allows for an alternative representation of the data that attempts to mitigate bias, retain as much of the original information as possible, and retain the mapping between the covariates and the outcome. However, in the same way as disparate impact removal, learning fair representations can only change the numerical attributes of the AIR data set. Figure 11 shows the process of implementing learning fair representations and training/testing the models.

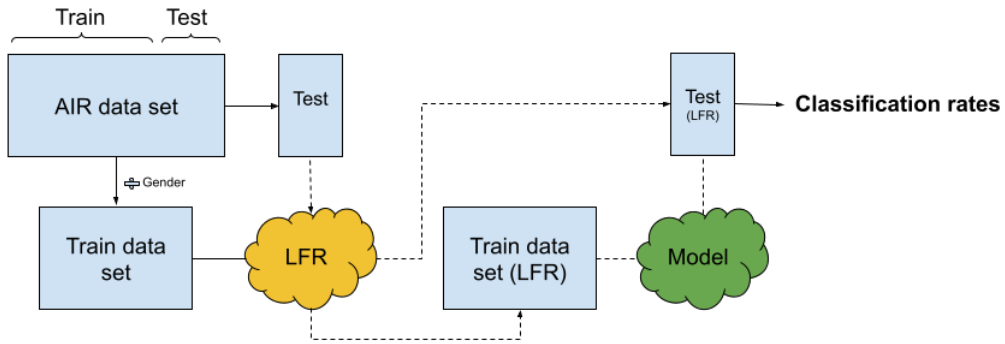


Figure 11: Learning Fair Representations

As can be seen in the figure, we drop gender from the training set and learn a fair representation of the training set. We use this learned representation to also transform the test into the new representation space. Then we train a model on the LFR training data set and test the model on the LFR test set.

We use an implementation of learning fair representations from the AIF360's pre-processing algorithm package [38].

Accuracy and output probabilities
When attempting to mitigate bias, one typically experiences a decrease in model performance since bias mitigation can be seen as a constraint to a model [19]. Therefore, when implementing a bias mitigation technique, we will compare the accuracy of the resulting models with models built on the original data set. This enables us to assess the impact that the bias mitigation technique has had on model performance.

We identify a discrepancy between our bias identification method and the intended use of the AIR model. When implemented, caseworkers in Aalborg Municipality will use the AIR model as a decision support tool. The model output will most likely be converted to some risk score (e.g., 0-100), directly mapped to the predicted probabilities of the model. The risk score will then be used to assess each citizen individually as described in the AIR case presentation in section 6. However, our method for identifying bias uses the predicted binary classes (fall and no fall), not the predicted probabilities. Therefore, it is relevant to assess what effect the bias mitigation technique has had in terms of the predicted probabilities.

Because of this, we will compare accuracy and predicted probabilities of models built on data sets where bias mitigation has been applied, with original models built on the original data set.

8.3 Machine learning models built on the AIR data set

We implement five machine learning models on the AIR data set: support vector machine (SVM), logistic regression (LR), random forest (RF), feed-forward neural network (FFNN), and XGBoost. The first four (SVM, LR, RF, and FFNN) are chosen by us, while XGBoost is the model that the AIR project team has chosen to build. We choose to test the bias identification and mitigation techniques across all five models. This is done to ensure robustness of our results.

The four models (SVM, LR, RF, and FFNN) are chosen since all of them are frequently used for classification. They are highly accessible, in the sense that many open-source programming languages have libraries with implementations of them. This is desirable, since our ambition is that our results can be useful for other projects that evaluate bias in the public sector. Finally, they represent four different approaches to building machine learning models. Logistic regression represents a regression model, support vector machine represents instance-based algorithms, random forest is a tree-based ensemble method, and the feed forward neural network represents a deep learning model [39]. The fact that they come from different machine learning approaches can be seen as a way to test the identification and mitigation techniques in a more robust fashion.

This section briefly describes each model, including the underlying theory and our specific implementation of the models. In the AIR project, the outcome of interest (Fall) is a binary variable. Therefore, the section covers theory related to binary classification.

Logistic Regression
Logistic regression is a classification algorithm, where the output can be interpreted as posterior probabilities of belonging to a positive class.

Logistic regression uses a function $h_\theta(x)$ that is restricted to output values between 0 and 1, which sum to 1 over the different classes, in this case two classes, by using the sigmoid function [40, p. 119]:

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}} \tag{32}$$

The θ are learned parameters governing the mapping from input variables to the predicted output, and are used to predict the class a of new observation. In this sense, the probability of belonging to a class given the values of the input variables and parameterized by θ is: $h_\theta(x) = P(y = 1|x; \theta)$. Logistic regression models are fitted by maximizing the log-likelihood [40, p. 120]. In the binary setting, when optimizing the model, logistic regression uses the binary cross-entropy loss to evaluate its performance,

which is equivalent to maximizing the log-likelihood:

$$Cost(h_\theta(x), y) = -y \cdot \log(h_\theta(x)) - (1 - y) \cdot \log(1 - h_\theta(x)) \tag{33}$$

To optimize the parameters θ, gradient descent is used to find the global minimum of the convex loss function. Logistic regression is often used as a data analysis tool to understand the role of the independent variables in explaining the outcomes. Since each input variable can be given an individual θ-parameter, the contribution of input features can be compared to one another [40, p. 121].

For the logistic regression, we use the Scikit-learn library [41] implementation and choose to run the models with the default settings, except for the maximum number of iterations, where 1000 is used to achieve convergence [41]. We also set the parameter "probability" to True in order to generate probability outputs. Finally, the parameter class weight is set to "balanced" since it is an unbalanced problem (see section 9). We chose default values to ensure reproducibility and generality of our tests.

Support Vector Machine
The Support Vector Machine algorithm [42] can be used to classify binary data. The algorithm uses a Support Vector Classifier to separate the classes. The Support Vector Classifier is a soft margin classifier, meaning that it finds an optimal separating hyperplane that allows misclassifications [40, p. 419]. Because the classifier allows misclassifications, the algorithm is robust to outliers and overlapping classifications. The optimal margin is found through cross-validation.

The Support Vector Machine algorithm takes a relatively lower-dimensional data set and calculates the relationship between observations in a relatively higher dimension and finds a Support Vector Classifier that can separate the classes in this higher dimensional representation [40, p. 423]. It does this since finding an optimal separator for some distributions is not possible in lower dimensions. To predict the class of a new observation, SVM maps the observation into the relatively higher space, and the class is given by where the observation lies in relation to the learned optimal separating hyperplane. To ease computation, the data is not actually transformed into higher dimensions, but the difference between data points are calculated "as if" the observations were in a higher dimension. This is called the "kernel trick". The Support Vector Machine uses different kernels to calculate the relationship between observations in higher dimensions [40, p. 424]. The Scikit-learn implementation uses the radial kernel as default [42]. The radial kernel is defined by the following:

$$e^{-\gamma(a-b)^2} \tag{34}$$

Where a and b are the coordinates of observations in the dataset and γ scales the influence that observations have on each other when calculating the relationships. The radial kernel calculates the relationship in infinite dimensions.

For the Support Vector Machine, we use the Scikit-learn library implementation and choose to run the models with the default settings, except for the parameter 'probability', which we set to True. Finally, the parameter class weight is set to "balanced".

Random Forest
Random Forest is a tree-based classification algorithm. The algorithm classifies a new observation by running the observation through a large number of decision trees and uses the aggregate classification to classify the observation. Each decision tree in the random forest is created in two steps: 1) creating a bootstrapped dataset by sampling with replacement from the original dataset, 2) creating a decision tree using a random subset of variables from the bootstrapped dataset [40, p. 588]. The number of decision trees in the random forest is a hyperparameter. The Out-Of-Bag dataset is used as a test set to calculate the Out-Of-Bag error, which estimates the accuracy of the random forest on new data [40, p. 593]. The accuracy is used to find the best number of variables to be used at each step when creating the decision trees in the random forest.

We use the Scikit-learn library implementation for the Random Forest and choose to run the models with the default settings. The default number of random variables at each step is the square root of

the total number of features, while the default number of decision trees in a random forest is 100 [43].

Feed Forward Neural Network
The goal of the FFNN is to approximate any function f*. The function, f*(x), maps the input x to a category, y. Information flows from x through multiple layers, which together compose many functions. The number of layers constitutes the depth of network. Each unit in the network can be interpreted as playing the role of a neuron, which receives an input (e.g., an element of a vector) and compute an activation value using an activation function [44, p. 164-165].

The forward information flow (forward propagation) produces a scalar cost using a cost-function (C). Back propagation is the backward flow of information about the cost, which the network uses in order to compute gradients [44, p. 189]. Back propagation works by finding the partial derivatives $\delta C/\delta w$ (gradients), where w is any weight in the network. The gradients are then used for updating the weights in the network and the network is optimized using gradient descent [45].

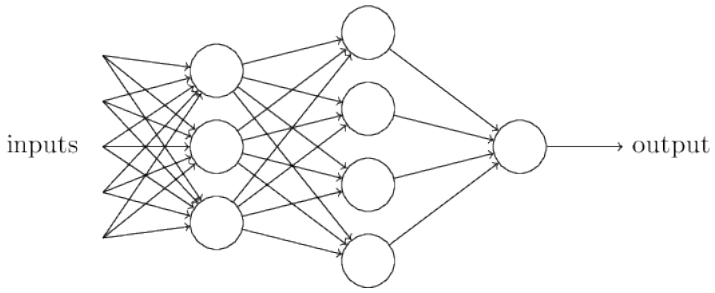


Figure 12: An example of a neural network from [45]

The linear transformation of each neuron's input is activated by an activation function, which yields a nonlinear transformation. We use the rectified linear activation function (ReLU) as activation function, since it is recommended as a default activation function [44, p. 170]. The advantage of ReLU is that the function eases optimization with gradient-based methods. The reason for this is that half of the domain outputs zero, which implies a more sparse network representation. Furthermore, whenever the unit is active, the ReLU function is linear and the gradient has a large constant value [44, p. 189]. The ReLU function is defined as:

$$g(z) = max(0, z) \tag{35}$$

where z=$W^T x + b$ and W describes the mapping (weights) of the the previous layer's output to the neuron. Each weight is represented by a edge in the graph. b is a bias term [44, p. 189].

The FFNN can be used for binary classification when the last output layer consists of one node where sigmoid is used as activation function. This gives output values between 0 and 1 [45], which can be interpreted as probabilities. The FFNN implemented using the AIR data is a fully connected network with an input layer, an output layer, and four hidden layers. The input layer and hidden layer use ReLU as an activation function. The input layer and hidden layers also use dropout and batch-normalization for regularization of the model. The output layer uses sigmoid as an activation function for binary classification. The loss function is the binary cross-entropy loss. The performance of the model is evaluated using cross-validation. After training, the FFNN can be used for prediction, by sending the features of a new observation to the input layer and propagating information forward to the output layer. PyTorch built-in libraries have been applied for the implementation of the model. The model architecture and the chosen hyper-parameters can be found in appendix, section L.

XGBoost
XGBoost is a library that contains an implementation of gradient-boosted decision trees [46]. Boosting involves training multiple models in a sequence and then letting the loss of a given model depend on the previous model's performance in the sequence [40, p. 653]. In Figure 13, boosting starts by building a model and then sequentially boost the performance by building new models [47]. Decision trees can be used as a boosting framework, where each model in the sequence is a tree [40, p. 654] [48].

Figure 13: Sequential combination of multiple models [47].

XGBoost (Extreme Gradient Boosting) is a machine learning system for tree boosting [48]. Tree models can be seen as a model that partition the feature (input) space into regions [49]. To classify a new observation, XGBoost uses K number of functions to predict the output, where each k function represent an independent tree [48]. The predicted outcome $\hat{y}$ of the input x_i is found by:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i) \tag{36}$$

where f_k corresponds to an independent tree. In other words, to make a classification, the input (x_i) enters each of the K trees, uses the decision rules of the tree to classify it into the leaves. The final prediction is calculated by summing up the score in the corresponding leaves.

In order to learn the f_t tree, two elements are used: 1) the losses of the previous tree (t-1), and 2) the prediction of the f_t tree itself [50]. First, the structure of the f_t tree is learned by considering every split on each feature. Then, the f_t tree predicts an output and adds the previous tree's output. A prediction at learning t step is then defined by:

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i) \tag{37}$$

The structure that is chosen for the t'th tree, is the one that minimizes the learning problem's objective function [49] [50]:

$$obj^{(t)} = l(\hat{y}_i, \hat{y}_i^{(t-1)}) + \sum_{a=1}^{t} \Omega(f_a) \tag{38}$$

Where l is a loss function. In a binary classification setting, the loss function is the binary cross-entropy loss [50]. The Ω is a regularization term that regularizes the complexity of the tree, for example, the number terminal nodes of each individual tree [49]. The objective function is optimized using gradient descent algorithm [51].

We use the settings as defined in the AIR XGBoost model found on the AIR project's GitLab [52] which also is found in appendix section L.

Cross-validation

In order to achieve estimates of the classification rates that are as robust as possible, we perform 5-fold cross-validation when training each classification model on the AIR data set. The 5-fold cross-validation is repeated ten times, each time with a new random seed. Therefore, each classification rate and accuracy has been calculated 50 times. For assessing the uncertainty of the models, the 95% confidence intervals are shown. The *central limit theorem* states that for a large number of model samples, the distribution of sampled means will approach a normal distribution [40, pp. 78-79]. We assume that the distributions of the metrics are normal and therefore, the confidence intervals are calculated accordingly. The confidence intervals show the range of values for each classification rate that with 95% confidence contains the true mean. If the confidence intervals of the estimate of the classification rates between females and males do not overlap, we deem it relevant to assess their relation and assess whether it is problematic according to the 80% rule.

Thresholds

The LR, RF, FFNN, and XGBoost use a classification threshold of 0.5, meaning that if the predicted probability of a citizen falling is less than 0.5, the binary prediction is "No Fall"; if above 0.5, the binary prediction is "Fall". Since the SVM does not output probabilities, the Scikit-learn implementation uses Platt scaling to produce probabilities by training a sigmoid function to map the SVM output to proba-

bilities [53] [54]. Therefore, SVM has no threshold on the probability outputs as the other four models. The optimal hyperplane is trained to divide the data into two groups: citizens who fall and not fall. The logistic regression is, after that, fitted to SVM outputs to generate predicted probabilities.

9 Descriptive Analysis

Before identifying bias in the classifications of the AIR model and our model implementations on the AIR data set, we will perform an initial descriptive analysis of the AIR data set to gain some domain knowledge that could be useful when assessing the classification rates and how they differ between groups. We will start by showing the distributions and averages of central variables to get an overall understanding of the data and try to understand how the variables might influence the probability of falling. We briefly go through descriptive statistics on the variables of the AIR data set. When showing the histograms, we have truncated the span of values shown since some of the bars in the histograms could be attributed to one single observation and showing this would not be in compliance with the data protection contracts we have signed.

Table 4 below shows how many of the citizens in the data set have experienced a fall within the first three months of their DigiRehab program. In the table, one can see that approximately 1 out of 5 citizens experienced a fall. This makes the prediction of falling an unbalanced problem, in the sense that a naive model which only predicts no citizens will fall could achieve fairly high accuracy of 77.7%.

Fall	**Count (%)**
Has not fallen	1666 (77.7%)
Has fallen	478 (22.3%)

Table 4: Distribution of Fall

Table 5 shows how many females and males there are in the data set. Females account for 2/3 of the data set, while males account for 1/3. The uneven distribution of gender could be because women on average live longer than men [55] and that the AIR data population is primarily elderly citizens.

Gender	**Count (%)**
Female	1365 (63.6%)
Male	779 (36.4%)

Table 5: Distribution of Gender

Table 6 shows that a higher proportion of the males experience falling compared to females. However, 284 females in the data set have fallen, compared to 194 males, which is explained by the fact that women are over-represented, and therefore the proportion of females that fall is less than the proportion of males that fall. We do not imagine a direct relationship between gender and falling, but rather that gender is correlated with other variables (observed and unobserved) that have a causal impact on falling, for example, physical impairments and inactive lifestyles.

Gender	**Count (%)**
Females falling	284 (20.8%)
Males falling	194 (24.9%)

Table 6: Distribution of citizens that falls grouped by gender

Figure 14 shows a histogram of the variable *Age*. As expected, the data set contains elderly citizens, most of whom are 70-100 years old, The mean age is 83.1. This validates our previous assumption of an elderly population, where the consequences of falling, and therefore also the consequences of a false negative, are quite severe. Finally, we imagine that the probability of falling increases with age.

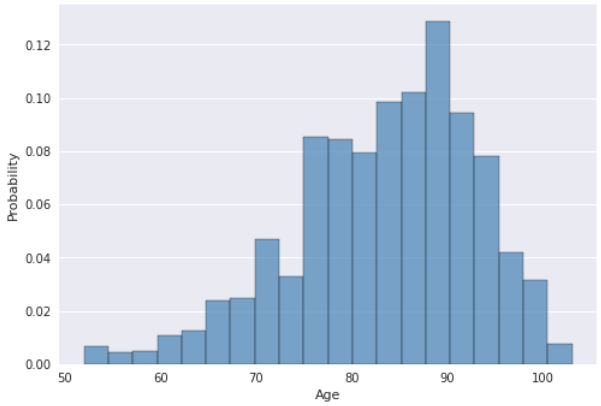


Figure 14: Histogram of age

Figure 15 shows the distribution of the number of aids. The distribution is right-skewed with a top of around five aids and a mean of 11 aids. When looking at the raw data, we observe that citizens do not need to have an aid to be part of the data; in fact, 6% do not have any aids. We expect that number of aids is positively correlated with the probability of falling since it implies some underlying physical impairment, which, all things equal, most likely heightens the risk of falling.

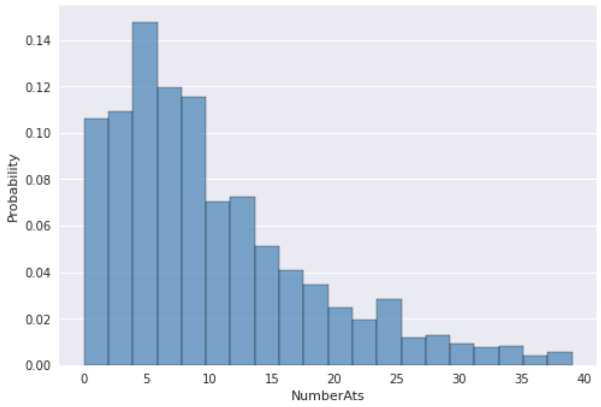


Figure 15: Histogram of number of aids

Figure 16 shows the distribution of loan period, which is the number of days a citizen (on average) has had aids on loan. The mean number of loan days is 1152. The distribution is right-skewed with a spike for very short loan periods and a plateau for loan periods between 1-3 years, after which the distribution is monotonically decreasing. The loan period is expected to be positively correlated with the probability of falling since it shows a lasting need for some aid.

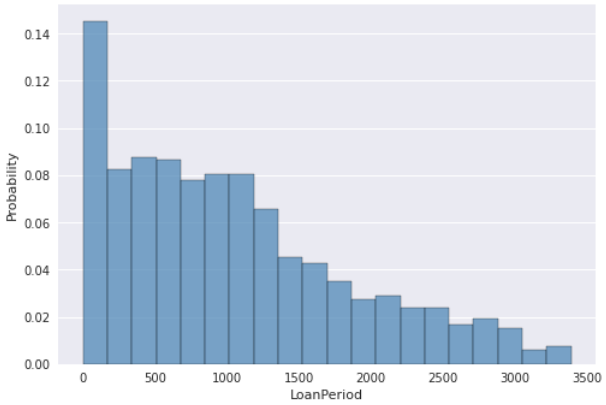


Figure 16: Histogram of loan period (days)

Figure 17 shows the number of citizens that belong to each of the clusters. The two clusters with most citizens are cluster 0 and 10.

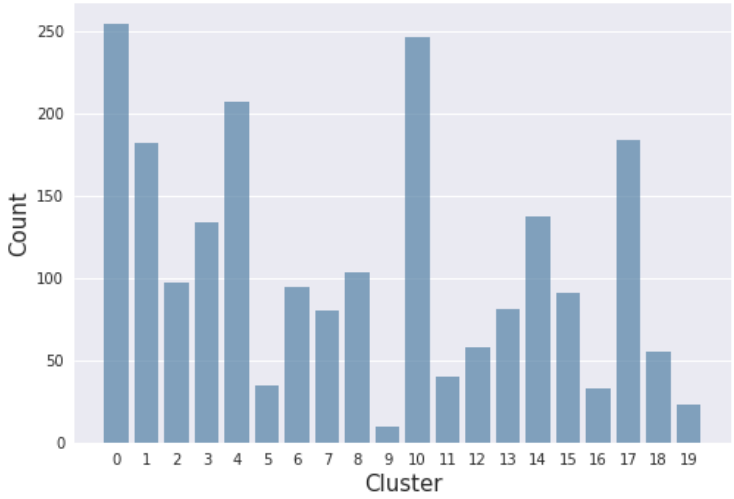


Figure 17: Count of citizens in each cluster

Table 7 shows the top five most loaned aids respectively for the citizens who do not fall and those who fall. The five most frequently loaned aids are the same for both groups, while a slightly higher percentage of those who fall have loaned the top five most loaned aids.

Aid	**Count "No Fall" (%)**	**Count "Fall" (%)**
Walker	1312 (78.8%)	406 (84.9%)
Shower chair	1193 (71.6%)	361 (75.5%)
Emergency alarm	855 (51.3%)	314 (65.7%)
Toilet seat riser	823 (49.4%)	246 (51.5%)
Bedsore prevention	652 (39.1%)	204 (42.7%)

Table 7: Top five most loaned aids.

Distribution of variables by gender
The purpose of the thesis is to identify and mitigate bias in relation to gender in the AIR project (see section 6.5.4). To learn more about potential differences in the distribution of central variables between females and males, we assess boxplots of *age*, *loan period*, and *number of aids*.

The distributions of age are in figure 18 plotted as box plots. The group of citizens who fall (the two rightmost distributions) have higher median ages than the group of citizens who do not fall (the two

leftmost). The group with the lowest median age is males who do not fall, implying that age might to a higher degree have an effect on the probability of falling for men.

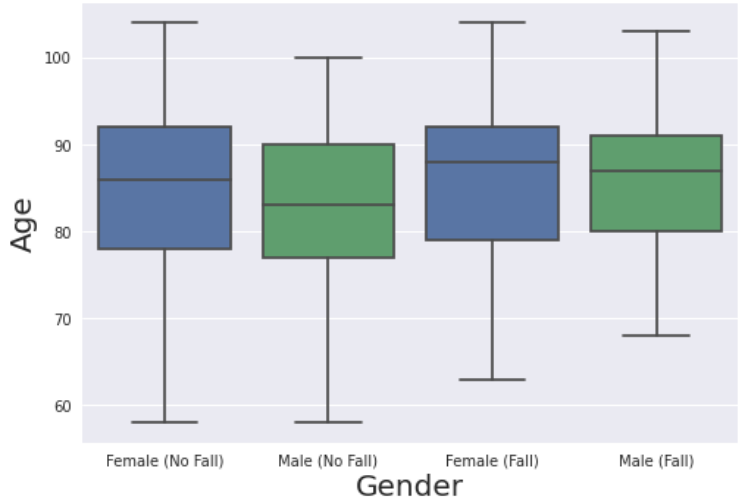


Figure 18: Box plot of the age grouped by gender and whether group members have fallen.

In figure 19, the median loan period is shorter for the citizens that fall for each gender. Furthermore, for the citizens who fall, males have shorter loan periods than females. These insights contradict our initial expectation, where we imagined that a long loan period would be related to a higher probability of falling. The opposite seems to be the case since 1) males (who fall at higher rates) have shorter loan periods and 2) those who fall have shorter loan periods than those who do not fall. If a short loan period maps to a high probability of falling, and males, in general, have shorter loan periods, this could impact that males are predicted as falling at a higher rate than females.

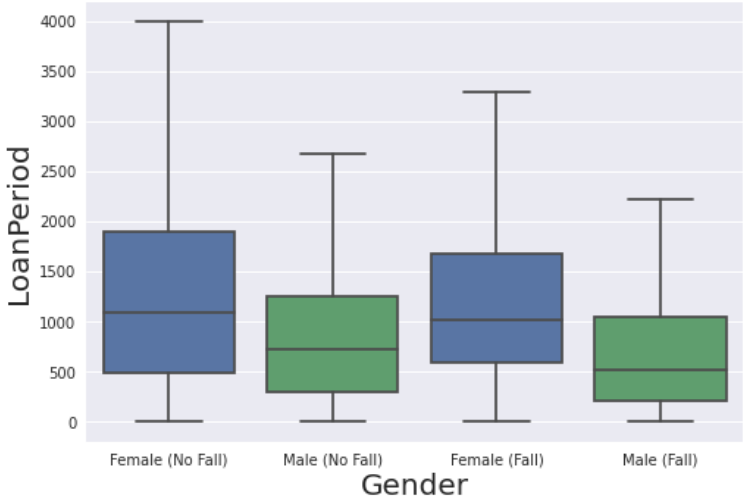


Figure 19: Box plot of loan period grouped by gender.

The distributions of the number of aids in figure 20 are more similar between the four groups in terms of medians and interquartile ranges. This contradicts our expectation that a higher number of aids leads to a higher probability of falling. The box plot in figure 20 therefore leaves a somewhat unclear relation between the number of aids and the probability of falling, where neither a positive or negative relation can be read from the plot.

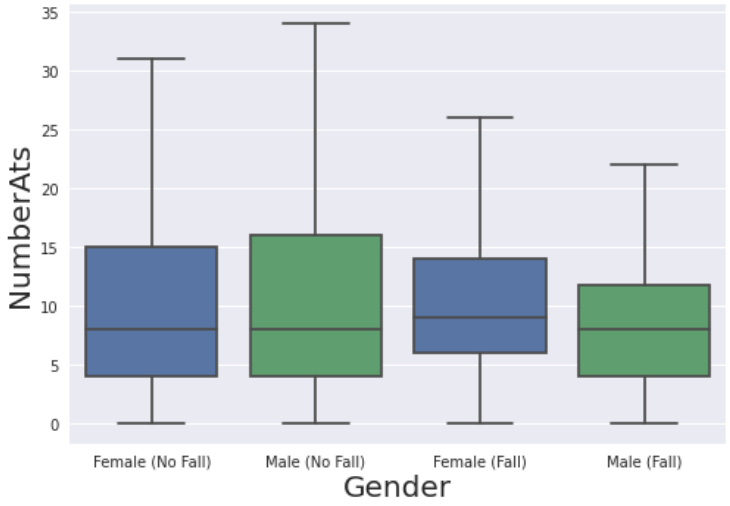


Figure 20: Box plot of number of aids grouped by gender.

Correlation between variables by gender
We briefly present a correlation matrix for each gender between the numerical variables and the fall-outcome. The information from these matrices could provide an additional understanding of the relation between the variables for females and males and how these might differ. Table 8 shows the correlations for females, and table 9 shows the correlations for males.

Variable	**Age**	**Loan period**	**Number of aids**	**Fall**
Age	1.0	-	-	-
Loan period	0.0637	1.0	-	-
Number of aids	-0.1522	0.1767	1.0	-
Fall	0.0235	-0.0268	0.0034	1.0

Table 8: Correlation matrix of numerical features and fall - females

Variable	**Age**	**Loan period**	**Number of aids**	**Fall**
Age	1.0	-	-	-
Loan period	-0.0937	1.0	-	-
Number of aids	-0.1977	0.3043	1.0	-
Fall	0.1114	-0.1113	-0.0843	1.0

Table 9: Correlation matrix of numerical features and fall - males

When looking at the differences between genders regarding the relation between the variable *Fall* and the other variables, the main difference is that the correlations seem to be stronger for males than females, which can be seen in the difference between the correlations of *Fall* and *Age* / *Loan period*. This could imply that the variables *Loan period* and *Age* affect the probability of falling for men to a higher degree than for women. Furthermore, *Fall* is positively correlated with *Number of aids* for females, while it is negatively correlated for males. However, the correlation for females is quite close to zero.

When looking at the difference between genders regarding the correlation between the variables other the *Fall*, one can see that *Loan period* and *Age* are positively correlated for females. In contrast, they are negatively correlated for males. This indicates that the relationship between age and loan period is different between females and males.

10 Identification of bias

For identifying bias in the models trained on the AIR data, we use a two-step strategy. First, we assess the classification rates across the five different algorithms and see if the estimates of the rates between females and males have overlapping confidence intervals. Second, we calculate the relation between the estimates of the classification rates and evaluate them in relation to the 80% rule. In other words, for identifying gender bias, we must both find that the confidence intervals of the classification rates do not overlap and that the relation between them is problematic according to the 80% rule.

10.1 Classification rates

In order to evaluate bias in the models used for classification, the four classification rates TPR, FPR, TNR, and FNR are estimated. These are shown in figure 21, while the values can also be assessed in the appendix, table 15 in section A. From figure 21, we can assess whether the confidence intervals of the classification rates for females and males overlap.

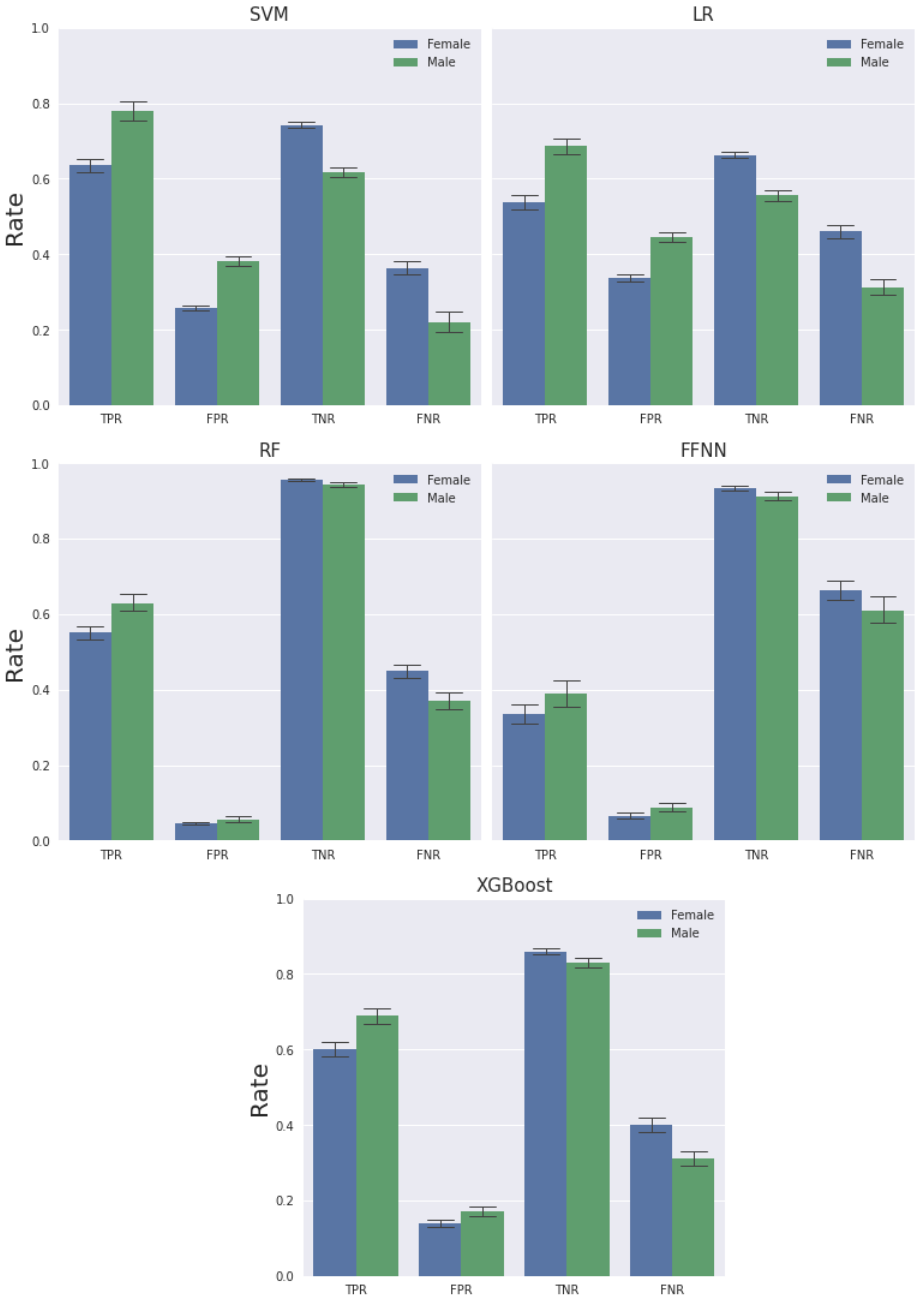


Figure 21: Classification rates of the five classification models grouped by gender.

Support Vector Machine (SVM)
In general, the SVM has higher positive classification rates for men (TPR and FPR) and higher negative classification rates (TNR and FNR) for women. This shows that the model accurately classifies males who actually have experienced a fall as "fallen" at a higher rate than women (TPR) and accurately classifies women who do not fall as "not fallen" at a higher rate than men (TNR). Similarly, the model misclassifies males who do not fall at a higher rate than females who do not fall (FPR), while also misclassifying females who fall at a higher rate than males who fall (FNR). As discussed previously, a misclassification in the form of a false negative has more severe consequences than a false positive misclassification. In this light, the classifier's potential bias is disadvantageous for women.

Logistic Regression (LR)
The LR exhibits the same pattern between the classification rates as the SVM. Again, LR wrongly classifies men at a higher rate in the positive "fall" class and wrongly classifies women in the negative "will not fall" class. Although for logistic regression, the false classification rates (FPR and FNR) are somewhat larger than for SVM.

Random Forest (RF)
For the RF implementation, there is a noteworthy change. First of all, the classifier almost makes no mistakes regarding the actual condition negative observations (TNR and FPR), that is, those citizens who do not fall, with an FPR around 0.05 and a TNR around 0.95. However, when assessing the TPR and the FNR, random forest has the same characteristics regarding gender as SVM and LR while having a slightly lower TPR and slightly higher FNR. This means that the classifier has worse performance on the actual condition positives, in other words, those who fall. The TPR and FNR show the same pattern regarding gender as SVM and LR.

Feed Forward Neural Network (FFNN)
The classification rates of the FFNN implementation resemble the RF regarding FPR and TNR. Furthermore, the FFNN model has the highest FNR and lowest TPR of all the implementations. Again, the same gender misclassification patterns exhibited by the other models can be found in the FFNN.

XGBoost
The XGBoost model exhibits a classification rate pattern somewhat in between that of the RF and FFNN, which are alike, and the SVM and the LR, which are alike. The FPR and TNR values are closer to the edge values (0 and 1) than SVM and LR but not as close to the edge as RF and FFNN. The TPR is higher than the FNR but not as exaggerated as is the case for SVM. Again, in the same way as the previous models, there is a gender misclassification pattern, where the FPR is higher for men and the FNR is higher for women.

We note here the fact that our implementations (SVM, LR, RF, and FFNN) "surround" the Aarhus University's implementation (XGBoost) in terms of classification rate pattern characteristics, which to a certain degree validates our attempt to choose algorithms for our implementations that cover a wide area of algorithms and potential classification and bias patterns to ensure the robustness of our experiments and results.

Overall differences in classification rate patterns for gender
When assessing the true classifications, the TPR and TNR, the TPR for all five models are lower for females than for males. This means that of the citizens who actually fall, the models' predictions are more accurate for male citizens than female citizens. The opposite is true in the case of TNR, which is higher for females than for males. This means that of the citizens who actually do not fall, the models' predictions are more accurate for female citizens than male citizens.

When assessing the false classifications, the FPR and FNR, we also identify a gender difference. Here, the FPR is higher for males than for females for each model. This means that the models more often suggest that males should be provided fall prevention training - even though they will not fall. The same is clear for the FNR, which is higher for females than for males, and shows that the models tend to wrongly predict that females do not fall - even though they actually will fall. As previously mentioned, a false negative has more severe consequences than a false positive in this particular case since not providing fall prevention training to someone who will fall is worse than providing the training to a citizen who

does not need it. Again, the classification rates reflect a disadvantage for females.

All of the estimates of classification rates for all models have non-overlapping confidence intervals between males and females, except for the FFNN model. For the FFNN model, the confidence intervals on FPR and TNR do not overlap, but do overlap on FNR and TPR. Since most of the models have non-overlapping confidence intervals between the genders for the classification rates, we move on to examine whether the gender-specific differences imply that the classifiers could be biased in relation to gender.

10.2 Relation between classification rates

To evaluate whether the non-overlapping estimates of the classification rates found above imply biased classifiers, we show the relation between the classification rates and assess them using the notion of the 80% rule. To calculate the relation of a classification rate, the estimate of the rate for females is divided by the estimate of the rate for males. This is shown in figure 22. If a classification rate is equal for both genders, the point lies on the red line (relation = 1.0). If a classificaion rate is higher for males than females, the point lies under the red line. If a classification rate is higher for females than males, the point lies above the red line. The grey lines represent the 80% region (0.8 and 1.25). If it is assumed that the 80% rule applies for all of the rates, a model could be biased if a point lies either above the upper grey boundary or below the lower grey boundary. This method enables us to compare the gender-specific differences across all classification rates.

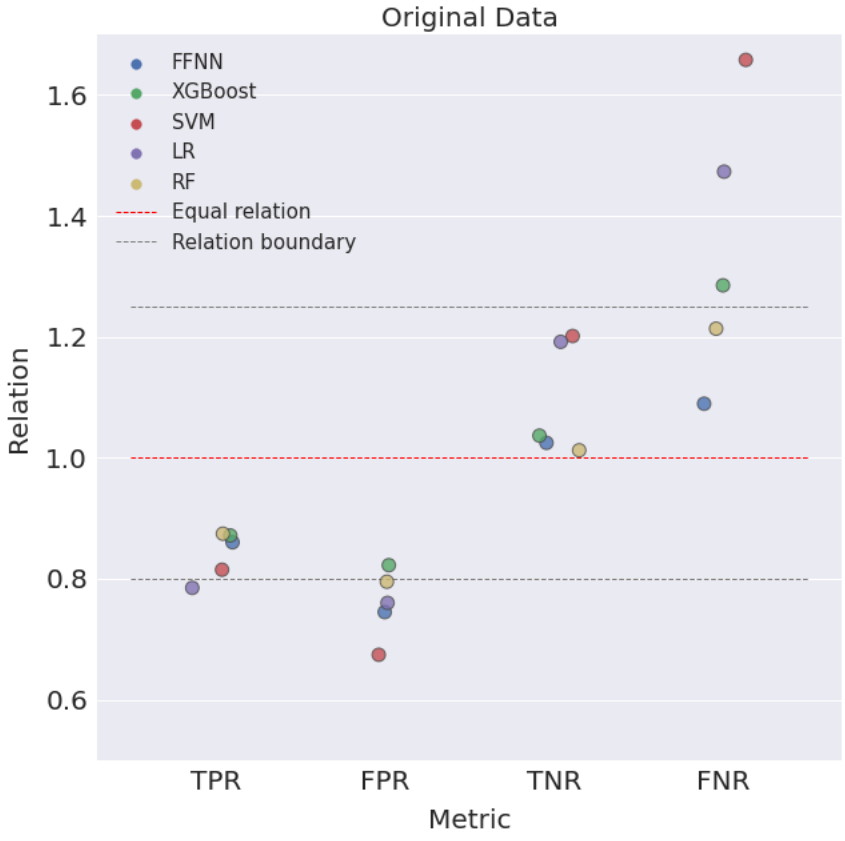


Figure 22: Relation between females and males in respectively TPR, FPR, TNR and FNR. The first axis is categorical. However, to avoid overlapping data points within the rates jitter is added to the plot.

Figure 22 shows what was already established in figure 21, which is that males have higher TPR and FPR and that females have higher TNR and FNR. However, figure 22 reveals insights into the relation between the classification rates and whether or not this entails a biased classifier.

For **TPR**, four out of five of the models lie within the 80% region, while the remaining, the LR, lies just outside.

For **FPR**, four models lie outside of the 80% region, while XGBoost is inside the region although close to the margin. Regarding FPR, the SVM implementation is the most problematic classifier.

For **TNR**, all relations between men and women lie within the 80% region. The SVM and LR are closer to the upper boundary than the rest of the models.

For **FNR**, the FFNN implementation is within the 80% region, while RF is closer to the margin but still within. Furthermore, the SVM and the LR implementations are clearly outside of the 80% region, and the XGBoost is also outside but closer to the boundary.

When looking across all rates, the results lead us to conclude that there could be potential issues with bias regarding the FPR and FNR when implementing models on the AIR data set. This bias is disadvantageous for females in the sense that males who do fall are more likely to be provided with fall prevention training (FPR) and that females who fall are less likely to be provided with fall prevention training (FNR). For TPR, the picture is less clear where the implementations are close to the margin of the 80% region, while it seems that the TNR is not biased.

When looking at the XGBoost model, which is the algorithm that Aalborg Municipality and Aarhus University intend to implement in the real world, it appears to be one of the less biased classifiers since the relation of the classification rate estimates are within or close to the acceptable 80% region for all rates. We have identified bias in the AIR classification algorithm (XGBoost) and seen that among four other algorithms trained on the AIR data the XGBoost method is one of the less biased. Furthermore, the XGBoost model's challenges with bias are limited to FPR and FNR, where it exhibits a gender bias that is close to the margins of the 80% region.

10.3 Accuracy and predicted probabilities

In order to assess the effect of a mitigation technique on the accuracy and predicted probabilities of the models, we show the values for the models trained on the original data set, to enable comparison.

The estimate of accuracy for each model is shown in figure 23.

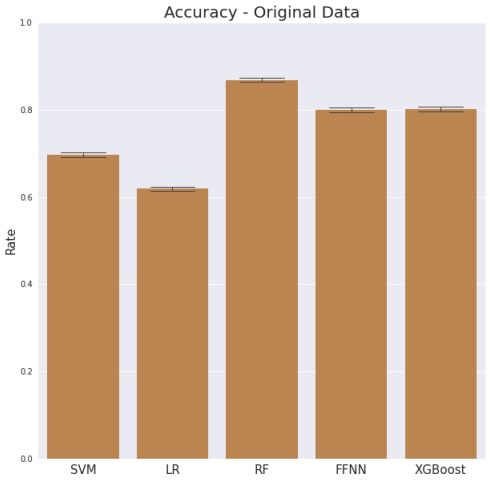


Figure 23: Accuracy of the five models.

In figure 23 we observe that the accuracies lie between 0.62 and 0.87. RF has the highest accuracy, followed by XGBoost and FFNN with a slight drop in accuracy. SVM is somewhat lower, while LR is clearly the lowest. The estimates can be found in table 16 in appendix A.

Table 10 shows the predicted probabilities for each model conditioned by gender and ground truth outcome.

Model	Females (Fall)	Males (Fall)	Females (No Fall)	Males (No Fall)
SVM	31.0	36.3	18.0	22.2
	(30.5-31.6)	(35.7-36.9)	(17.8-18.3)	(21.8-22.5)
LR	50.3	56.8	42.4	47.2
	(49.7-50.9)	(56.1-57.5)	(42.0-42.7)	(46.68-47.66)
RF	50.0	55.4	14.3	15.4
	(49.0-51.0)	(54.2-56.6)	(14.0-14.6)	(15.0-15.9)
FFNN	34.9	38.6	17.2	19.9
	(34.0-35.7)	(37.6-39.6)	(16.9-17.4)	(19.5-20.4)
XGBoost	54.9	59.5	20.7	24.0
	(53.8-56.0)	(58.2-60.8)	(20.2-21.1)	(23.3-24.6)

Table 10: Mean and 95% confidence interval for predicted probabilities conditioned by actual fall and gender.

As expected, since men fall more on average than women, males generally have higher predicted probabilities than females. We identified a potential bias, where the FNR for females is higher than males, and where the FPR for males is higher than females. Therefore, when assessing the predicted probabilities in the tested mitigation techniques, we hope to find, that the probabilities for females who fall increase (lower FNR), while the probabilities for males who do not fall decrease (lower FPR). Furthermore, we are less focused on changing the probabilities for males who fall and for females who do not fall, since the issues with bias are not directly related to the probabilities of these two groups.

10.4 Sub-conclusion: Identification of bias in AIR

On the basis of section 10 we have answered research question 1. We have identified bias in the AIR project in the following way:

In section 10.1, we identified bias by estimating the gender specific differences in classification rates across all five models. Here, we found non-overlapping confidence intervals, indicating gender specific differences regarding TPR, TNR, FPR, and FNR.

In section 10.2, we identified bias by assessing the relation between the classification rates of females and males and evaluated them in the light of the 80% rule. Here, we found that the relations between females and males in terms of FPR and FNR are biased. This is particularly the case for FNR, where females who fall are wrongly classified as not falling. This misclassification can have severe consequences.

In section 10.3, we assessed the predicted probabilities of all five models. Here, we observed, that males have higher predicted probabilities of falling than females.

11 Mitigation of bias

In this section we test four bias mitigation techniques. These are:

- Dropping the gender variable
- Gender swap
- Disparate impact removal
- Learning fair representations

We implement each of them, train the five models on the resulting data set, and assess if the identified bias regarding FPR and FNR has been mitigated.

First, classification rates of the new models are compared with the original rates from figure 21. Second, the relations of the new classification rates are compared with the relations of the original rates from figure 22. Third, the average predicted probabilities are compared with the originals' found in table 10.

11.1 Dropping the protected variable

We remove the gender variable from the AIR data set and train the models again.

11.1.1 Classification rates

Figure 24 shows estimates of the classification rates from models trained on gender swapped AIR data set. These are compared to the original plots from figure 21. The new estimates of classification rates can also be found in appendix, section A, table 20.

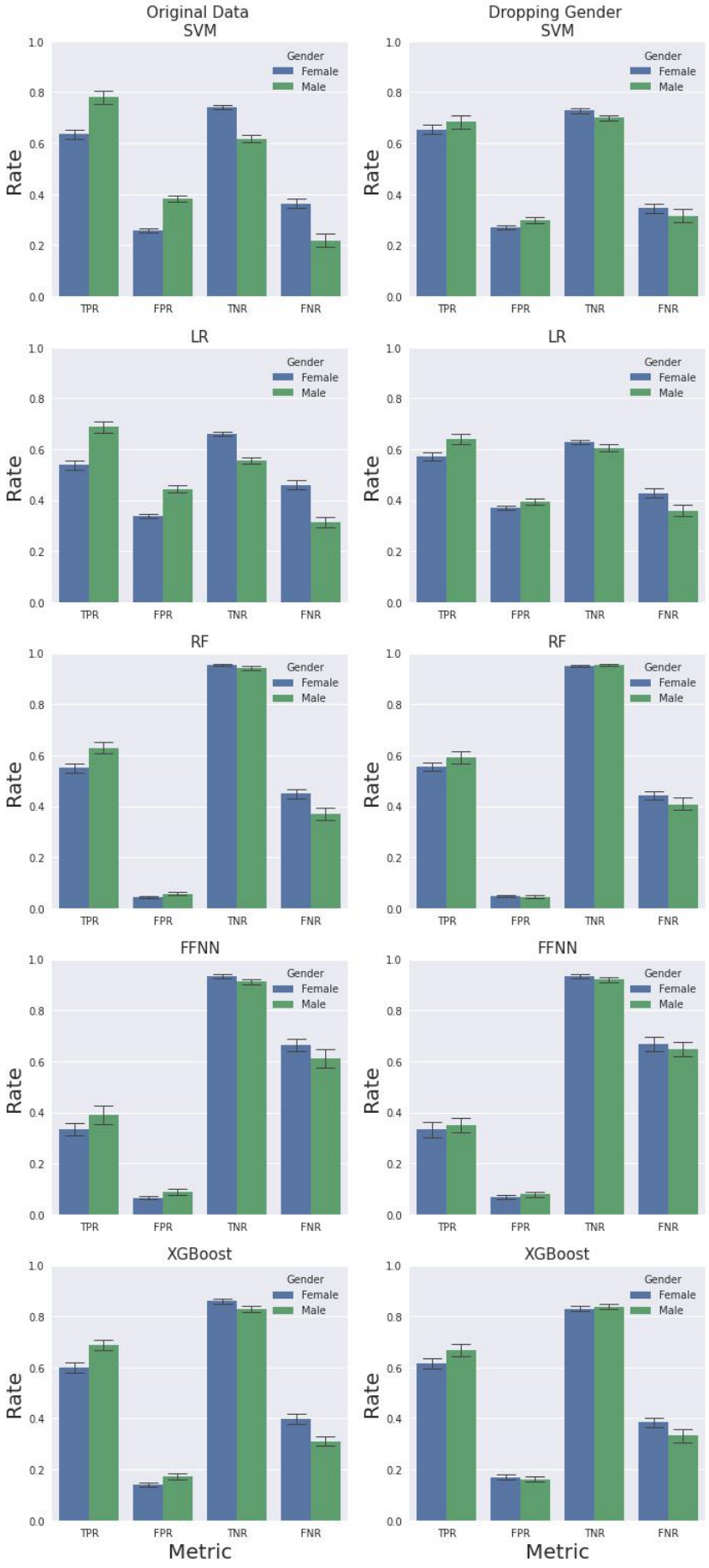


Figure 24: Comparing classification rates between models built on original data (left) and data where gender is dropped (right).

SVM
FPR decreases and FNR increases for males. The confidence intervals for FPR are still not overlapping, while the FNR confidence intervals overlap.

LR
Dropping the gender variable increases FPR for females and decreases FPR for males. Furthermore, it slightly decreases FNR for females and increases FNR for males. However, the confidence intervals of the estimates are non-overlapping for both FNR and FPR.

RF
For males, there is a decrease in FPR and an increase in FNR. The confidence intervals for both FPR and FNR overlap.

FFNN
Dropping gender decreases FPR and increases FNR for males. The confidence intervals for both FPR and FNR overlap.

XGBoost
FPR decreases for males and FPR increases for females, while there is an increase in FNR for males and a decrease in FNR for females. The confidence intervals for FPR overlap, while FNR confidence intervals do not overlap.

General comments
It seems that for all models, the difference between men and women regarding the identified FNR and FPR bias is decreased.

For SVM, RF, and FFNN, the FNR-related biases are mitigated by only increasing FNR for males. This implies that the performance of the models is worsened. Simply increasing FNR for males is an undesirable avenue for bias mitigation since, as previously stated, FNR is a costly misclassification. In the LR and XGBoost, dropping the gender variables lowers the FNR for females and increases the FNR for males. This is a somewhat more desirable result.

For all of the models, FPR for males is decreased, and FPR for females is mostly unchanged or slightly increased. This is a desirable mitigation result.

11.1.2 Relation between classification rates

In figure 25, the left pane shows the relations of metrics of the models built on the original data. The right pane shows the relations of the metrics for the models built on the data where gender is dropped.

The original models exhibited bias in FPR and FNR. This bias is now mitigated since all metric relations are located within the 80% region. Furthermore, the TPR and TNR are located closer to the red line (relation=1) than was the case with the models trained on the original.

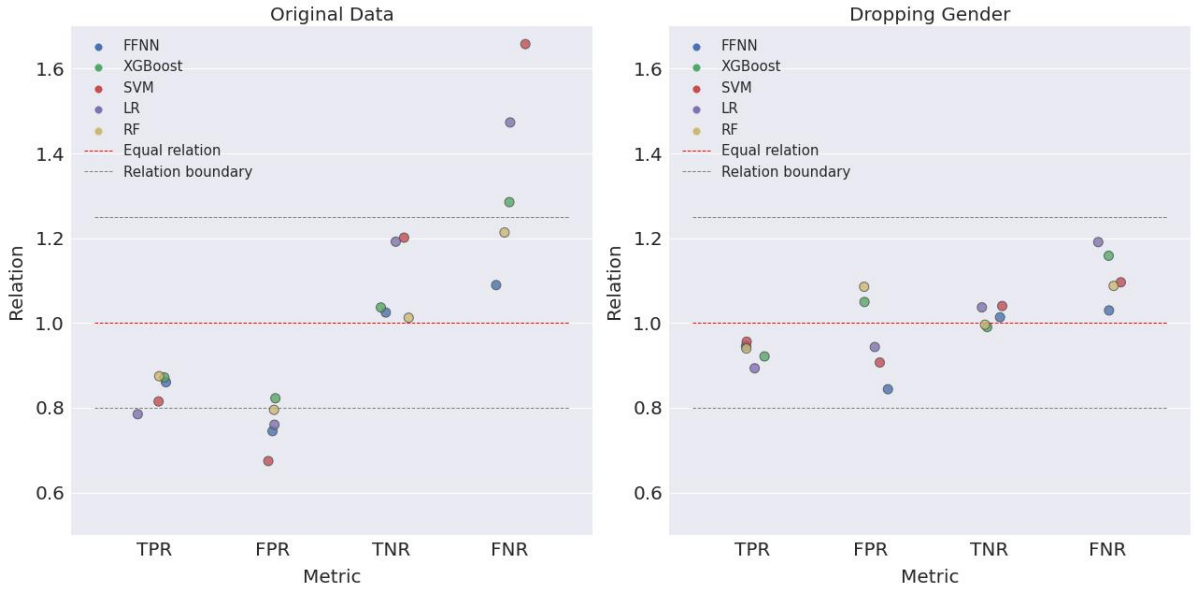


Figure 25: Relation of classification rates between males and females. Left: Models built on original data. Right: Models built on data where gender is dropped.

Thus, when dropping the gender variable and training the five models on the resulting data set, no bias related to the 80% rule is identified when using our approach for bias identification. This means that dropping the gender variable successfully mitigated the identified gender bias.

11.1.3 Accuracy and predicted probabilities

Figure 26 shows the accuracy of each model compared to the model trained on the original data set.

Figure 26: Accuracy of the five models when mitigating bias by dropping gender.

From figure 26 it is clear that dropping gender did not have a substantial impact on the accuracy of the models. The estimates can be found in table 18 in appendix A

Finally, we assess the averages of the predicted probabilities to see if dropping the gender variable has had the desired effect here also. Table 11 shows the changes in probabilities grouped by gender and actual outcome, compared with the original probabilities. As stated previously, we focus on increasing the probabilities for the group "Females (Fall)" and decreasing the probabilities for the group "Males (No Fall)".

Model	Females (Fall)	Males (Fall)	Females (No Fall)	Males (No Fall)
SVM	+0.6	-3.6	+1.4	-2.1
LR	+1.7	-2.7	+1.6	-2.8
RF	+0.2	-2.1	+0.7	-0.9
FFNN	+0.6	-2.2	+0.8	-0.9
XGBoost	+0.9	-1.5	+2.4	-0.1

Table 11: Mean change of predicted probabilities conditioned by actual fall and gender. From models trained on data where gender is dropped compared with original models.

We find a clear effect, where the probabilities of females are increased and the probabilities of males are decreased. This is a positive result. However, it seems that the probabilities of females who do not fall are increased more than probabilities of females who do fall. Similarly, the probabilities of males who fall, are decreased more than the probabilities of males who do not fall. This is unwanted, since the effect of the mitigation technique is skewed in an undesired direction, where we impact the "wrong" groups to a higher degree, in this case "Males (Fall)" and "Females (No Fall)".

11.2 Gender swapping

We concatenate the original data set with a gender swapped version of the data and train each model again. The resulting models are then tested on a test set from the original data.

11.2.1 Classification rates

Figure 27 shows estimates of the classification rates from models trained on gender swapped AIR data set. These are compared to the original plots from figure 21. The new estimates of classification rates can also be found in appendix, section A, table 20.

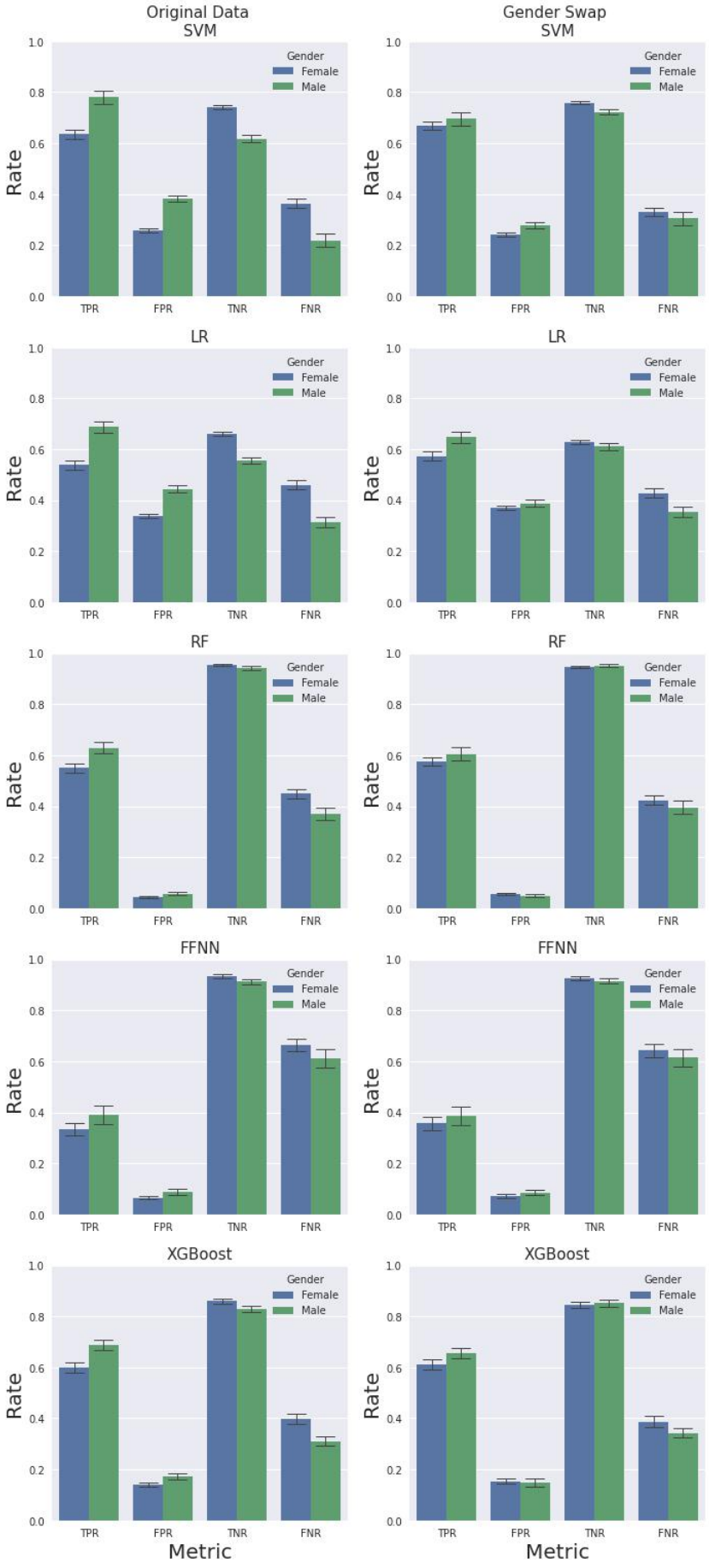


Figure 27: Comparing classification rates between models built on original data (left) and gender swapped data (right).

SVM
When training on a gender swapped data set, FPR decreases and FNR increases for males, while FNR slightly decreases for females. The confidence intervals for FNR do not overlap, while they overlap for FNR.

LR
FPR decreases for males and slightly increases for females. FNR decreases for females and increases for males. The confidence intervals overlap for FPR and do not overlap for FNR.

RF
For FPR, there is a slight increase for females and a slight decrease for males. FNR decreases for females, while it increases for males. The confidence intervals for both FPR and FNR overlap.

FFNN
There is no notable change in the levels of the classification rates after implementing gender swap. The confidence intervals overlap for FPR and FNR.

XGBoost
When mitigating bias using gender swap, FPR increases for females and decreases for males. FNR decreases for females and increases for males. The confidence intervals for FPR overlap and do not overlap for FNR.

General comments
It seems that for most of the models, the difference between men and women regarding the identified FNR and FPR bias is decreased. This is primarily done by decreasing FPR and increasing FNR for males and increasing FPR and decreasing FNR for females. In other words, more males are predicted as not falling, and more females are predicted as falling, compared to the original models. When comparing the effect of bias mitigation through gender swapping with dropping gender, we identify an important difference regarding FNR. For gender swapping, the FNR values for females and males move towards each other instead of simply increasing the FNR values for males only. Gender swapping is a better way to mitigate bias since the overall number of false-negative is lower for gender swap than dropping gender. This is a desirable result since false negatives are costly for elderly citizens.

The confidence intervals for FPR and FNR overlap for eight out of ten model-specific classification rates (FPR and FNR for each model).

11.2.2 Relation between classification rates

In figure 28, the left pane shows the relations of metrics of the models built on the original data. The right pane shows the relations of the metrics for the models built on the gender swapped data.

The original models exhibited bias in FPR and FNR. This bias is now mitigated since all metric relations are located within the 80% region. Furthermore, the TPR and TNR are located closer to the red line (relation=1) than was the case with the models trained on the original.

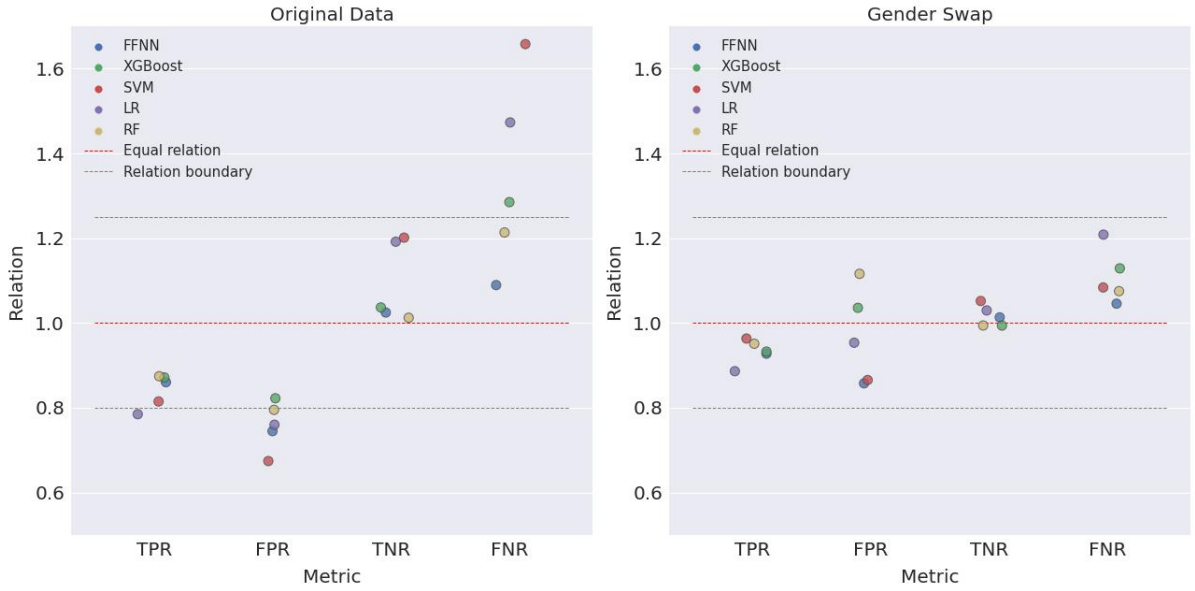


Figure 28: Relation of classification rates between males and females. Left: Models built on original data. Right: Models built on data where gender swap is implemented.

Thus, when mitigating bias using gender swapping, no bias related to the 80% rule is identified. This means that gender swapping successfully mitigated the identified gender bias.

11.2.3 Accuracy and predicted probabilities

Figure 29 shows the accuracy of each model trained on the gender swapped data set, compared with the original models.

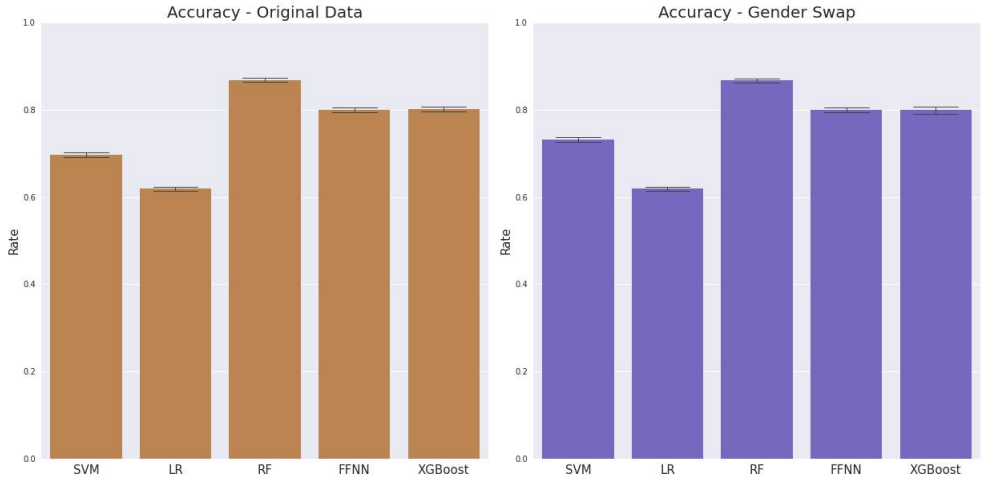


Figure 29: Accuracy of the five models when mitigating bias using gender swapping.

From figure 29 it can be seen that gender swapping does not have any noteworthy effect on the accuracy of the models. The estimates can be found in table 21 in appendix A.

Finally, we assess the averages of the predicted probabilities to see if gender swap has had the desired effect here also. Table 12 shows the changes in probabilities grouped by gender and actual outcome, compared with the original probabilities.

Model	Females (Fall)	Males (Fall)	Females (No Fall)	Males (No Fall)
SVM	+7.4	+3.4	-1.1	-4.2
LR	+1.8	-2.4	+1.2	-3.3
RF	+5.1	+2.0	-0.6	-2.3
FFNN	+0.2	-1.9	+0.6	-1.3
XGBoost	+1.0	-1.6	+0.7	-2.2

Table 12: Mean change of predicted probabilities conditioned by actual fall and gender. From models trained on gender swapped data compared with original models.

When assessing the predicted probabilities, we find that the effects are as desired. The females who fall experience the largest increase, while the males who do not fall experience the largest decrease in predicted probabilities. Furthermore, the impact on the groups "Males (Fall)" and "Females (No Fall)" is unclear, where the direction of changes in probabilities depends on the model used for prediction. Overall, the changes in predicted probabilities are desired.

11.3 Disparate impact removal

We perform disparate impact removal (DI removal) on the numerical features in the training data set, which are *age*, *loan period*, and *number of aids*. The models are trained on the training data set and tested on data set where disparate impact removal has also been performed.

11.3.1 Classification rates

Figure 30 shows estimates of the classification rates from models trained on disparate impact removed AIR data set. These are compared to the original plots from figure 21. The new estimates of classification rates can also be found in appendix, section A, table 23.

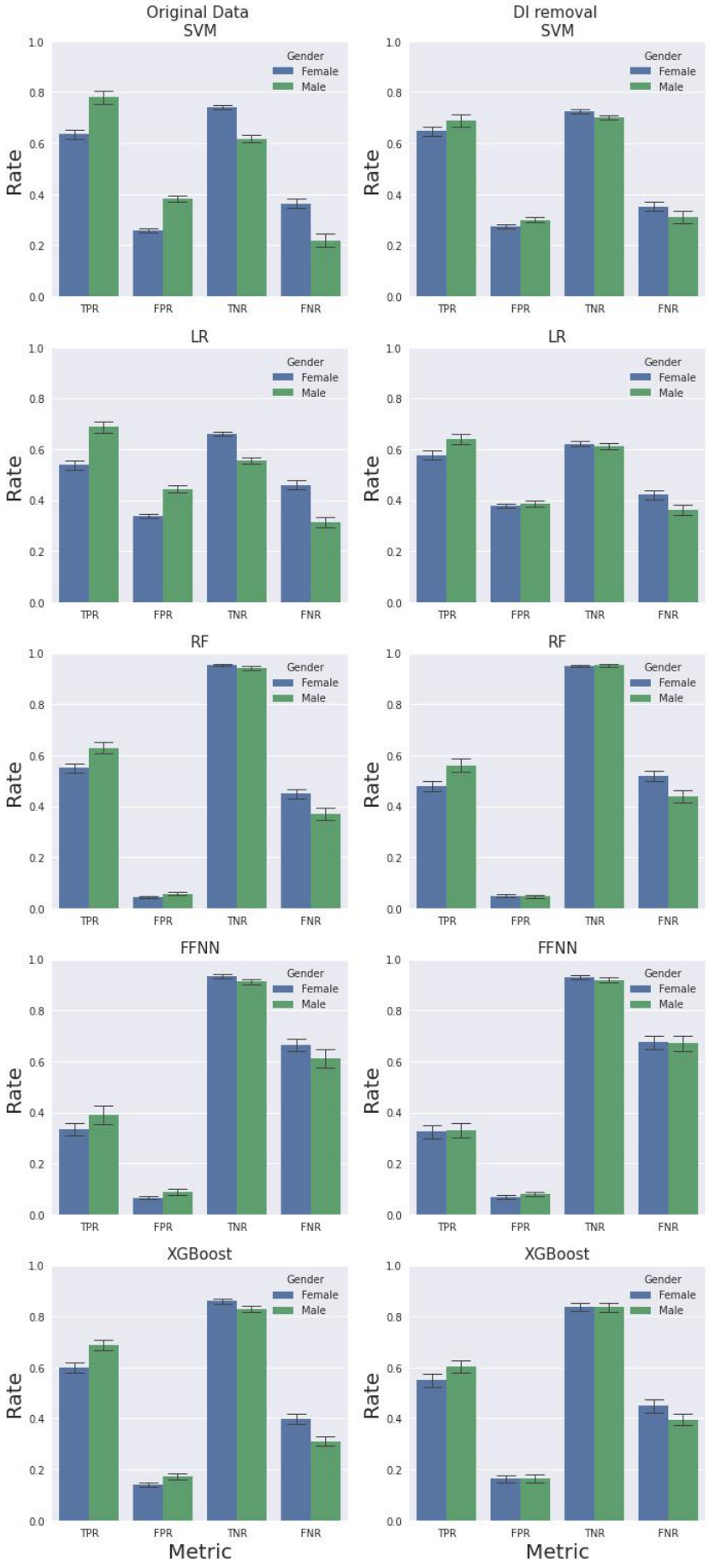


Figure 30: Comparing classification rates between models built on original data (left) and data that has been DI removed (right).

SVM
FPR decreases and FNR increases for males. The confidence intervals for FPR do not overlap, while the FNR confidence intervals overlap.

LR
DI removal increases FPR for females and decreases FPR for males. Furthermore, it slightly decreases FNR for females and increases FNR for males. FPR confidence intervals overlap. However, the confidence intervals of the estimates are non-overlapping for FNR.

RF
For males, there is a decrease in FPR and an increase in FNR. For females the FNR increases. The confidence intervals for FPR overlap, but FNR does not overlap.

FFNN
DI removal decreases FPR and increases FNR for males. The confidence intervals for FPR do not overlap, while they do overlap for FNR.

XGBoost
FPR increases for females, while there is an increase in FNR for males and for females. The confidence intervals for FPR and do not overlap for FNR.

General comments
It seems that for all models, the difference between men and women regarding the identified FNR and FPR bias is decreased. However, for the RF and XGBoost, FNR increases for both females and males, which is not a desired result. Five out of ten confidence intervals for FPR and FNR do not overlap.

11.3.2 Relation between classification rates

In figure 31, the left pane shows the relations of metrics of the models built on the original data. The right pane shows the relations of the metrics for the models built on the disparate impact removed data.

The original models exhibited bias in FPR and FNR. This bias has been mitigated since all metric relations are located inside the 80% region. Furthermore, the TPR and TNR are located closer to the redline (relation=1) than was the case with the models trained on the original.

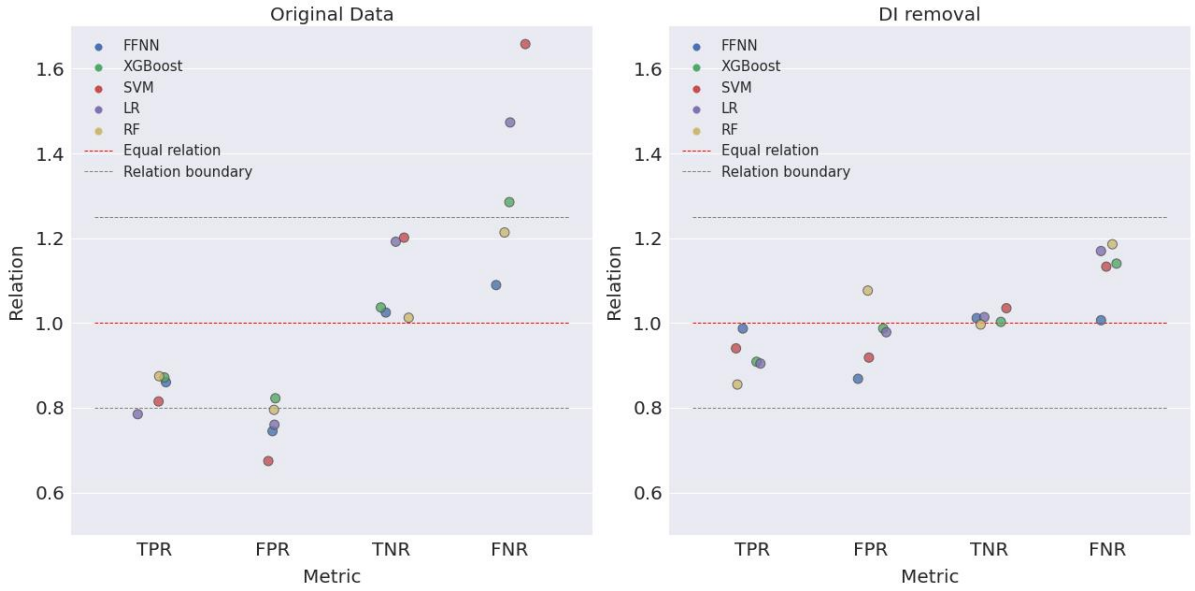


Figure 31: Relation of classification rates between males and females. Left: Models built on original data. Right: Models built on data where DI removal is implemented.

Thus, when creating a DI-removed version of the data set and training the five models on the resulting data set, the bias is mitigated successfully.

11.3.3 Accuracy and predicted probabilities

Figure 32 shows the accuracy of each classification algorithm, compared with the original.

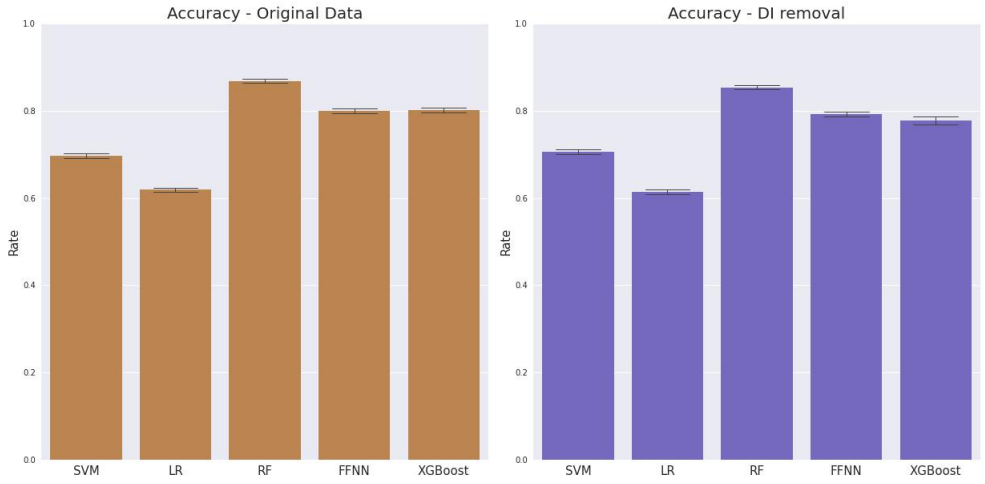


Figure 32: Accuracy of the five models when mitigating bias with DI removal.

From figure 32 it can bee seen that training on a DI-removed data set, did not have any noteworthy impact on the accuracy of any model.

Finally, we assess the averages of the predicted probabilities to see if disparate impact removal has had the desired effect here also. Table 13 shows the changes in probabilities grouped by gender and actual outcome, compared with the original probabilities.

Model	Females (Fall)	Males (Fall)	Females (No Fall)	Males (No Fall)
SVM	+0.5	-3.3	+1.4	-2.0
LR	+2.0	-2.7	+2.0	-3.2
RF	-5.0	-5.7	+1.5	-0.2
FFNN	+0.5	-2.9	+1.4	-0.9
XGBoost	-4.3	-5.3	+3.2	0

Table 13: Mean change in predicted probabilities conditioned by actual fall and gender. From models trained on DI removal data compared with original models.

When assessing the table, we find the clearest effect for the groups "Males (Fall)" (decrease) and "Females (No Fall)" (increase). This is not a desired result. For "Females (Fall)", the results are somewhat unclear, where three models indicate a slight increase in probabilities, while the remaining two indicate a notable decrease. For "Males (No Fall)", the picture is clearer, where four of five models indicate a decrease (as desired). However, the decrease is less substantial than for "Males (Fall)", which is not a desired result.

11.4 Learning fair representations

We find a fair representation of the numerical features of the data set using the training data. The same transformation is afterward applied to the test set. Each model is trained on the LFR training data and tested on the LFR test data.

11.4.1 Classification rates

Figure 33 shows estimates of the classification rates from models trained on LFR data set. These are compared to the original plots from figure 21. The new estimates of classification rates can be found in appendix, section A, table 26.

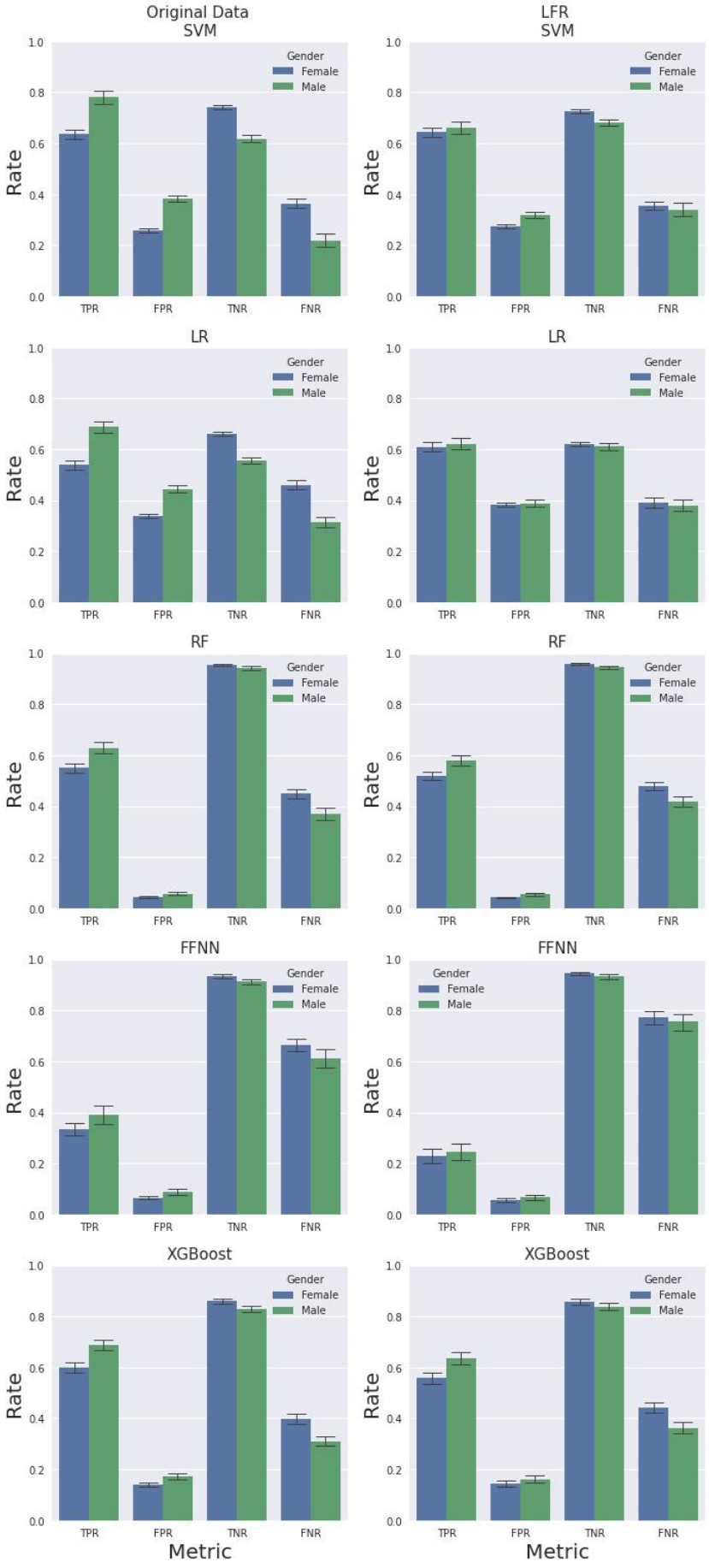


Figure 33: Comparing classification rates between models built on original data (left) and the LFR data (right).

SVM
When training on an LFR data set, FPR decreases and FNR increases for males, while FNR slightly decreases for females. The confidence intervals for FNR overlap, while they do not overlap for FPR.

LR
FPR decreases for males and increases for females. FNR decreases for females and increases for males. The confidence intervals overlap for both FPR and FNR.

RF
FNR increases for females and males. The confidence intervals for FPR overlap, but not for FNR.

FFNN
FPR decreases for females and males. FNR increases for both groups. The confidence intervals overlap for FPR and FNR.

XGBoost
When mitigating bias using LFR, FPR increases for females and decreases for males. FNR increases for females and males. The confidence intervals for FPR overlap and do not overlap for FNR.

General comments
For most of the models, the difference between men and women regarding the identified FNR and FPR bias is decreased. This is primarily done by decreasing FPR and increasing FNR for males. The changes are less consistent for females, where we identify mostly unchanged FPR and examples of both an increase and a decrease in FNR. However, as the only model, LR exhibits the desired results, where the FPR and FNR of males and females move towards each other. Using LFR seems to decrease the FNR difference between genders, but does so by increasing the FNR for both genders. This is an undesirable mitigation result since FNR in the setting of the AIR case is the most costly misclassification.

11.4.2 Relation between classification rates

In figure 34, the left pane shows the relations of metrics of the models built on the original data. The right pane shows the relations of the metrics for the models built on the LFR data.

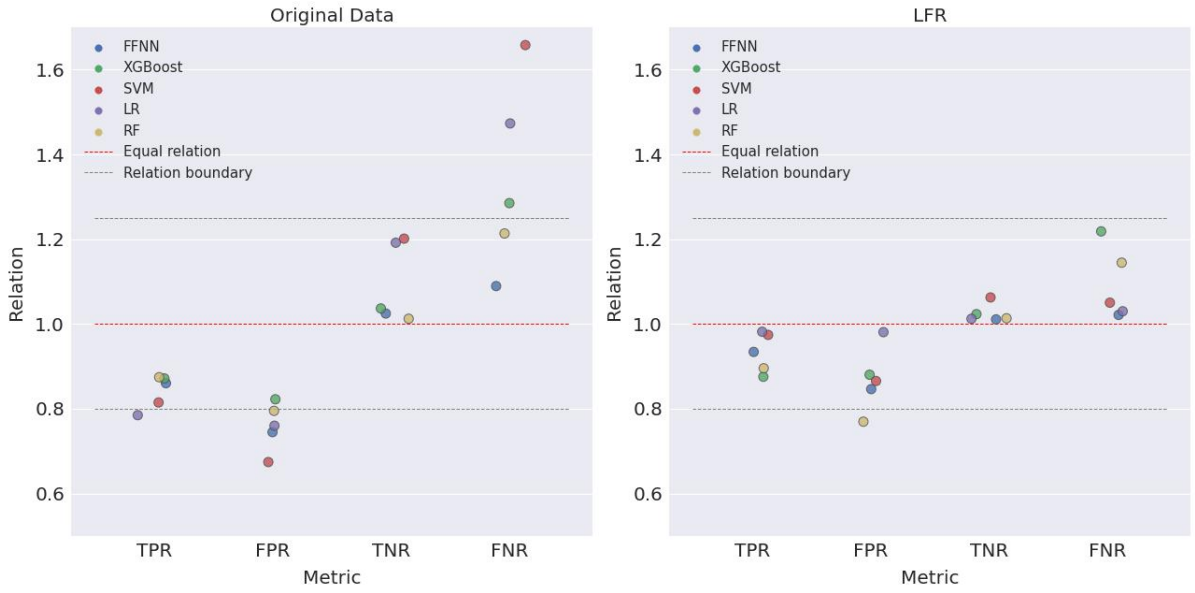


Figure 34: Relation of classification rates between males and females. Left: Models built on original data. Right: Models built on LFR data.

The original models exhibited bias in FPR and FNR. This bias is now mitigated since all metric relations are located within the 80% region - all but the FPR for RF. However, the confidence intervals for the

RF's FPR did overlap. Furthermore, the TPR and TNR are located closer to the red line (relation=1) than was the case with the models trained on the original.

Thus, when mitigating bias using LFR, no bias related to the 80% rule is identified. This means that LFR successfully mitigated the identified gender bias. However, we identify an undesired increase in FPR and FNR.

11.4.3 Accuracy and predicted probabilities

Figure 35 shows the accuracy of each model trained on the LFR data set, compared with the original models.

Figure 35: Accuracy of the five models when mitigating bias with LFR.

From figure 35 it can be seen that there is no noteworthy affect on the accuracy of the models, when LFR is used to mitigate bias. The estimates can be found in table 27 in appendix A.

Finally, we assess the averages of the predicted probabilities to see if learning fair representations has had the desired effect here also. Table 14 shows the changes in probabilities grouped by gender and actual outcome, compared with the original probabilities.

Model	Females (Fall)	Males (Fall)	Females (No Fall)	Males (No Fall)
SVM	+0.2	-4.5	+1.6	-1.8
LR	+2.1	-3.9	+2.2	-2.8
RF	-2.8	-2.8	+0.7	+0.5
FFNN	-2.1	-4.7	+2.1	+0.3
XGBoost	-2.8	-2.9	+1.9	+0.5

Table 14: Mean change in predicted probabilities conditioned by actual fall and gender. From models trained on LFR data compared with original models.

When assessing the table, the conclusions regarding the effect on predicted probabilities are similar to disparate impact (see table 13). The clearest effect is on the probabilities for "Males (Fall)" (decreases) and "Females (No Fall)" (increases). Furthermore, it seems that the models - despite not agreeing - indicate a decrease in probabilities for "Females (Fall)", while the picture is less clear for "Males (No Fall)". Overall, the impact on the predicted probabilities is not as desired.

11.5 Changes in overall classifications after mitigating bias

When commenting on each mitigation technique, we assessed changes for the gender-specific classification rates. However, to arrive at a conclusion on the effect of the mitigation techniques, we must also look at the effect on overall classifications. If we do not do so, we run the risk of missing important changes in classifications, for example, an overall increase in false negatives, which cannot be read only from the gender specific FNR. In this section, we assess what effects the different bias mitigation techniques have had on the classifications of each model. To this end, we show the values from a normalized confusion matrix in the appendix A, table 30. The values of each category (TP, FP, TN, and FN) are normalized, so they sum to 100. As previously mentioned, more false negatives are an undesirable effect of bias mitigation, while more false positives are less problematic. We briefly consider the changes of each model and mitigation technique compared to the original confusion matrix to conclude on the overall effects that mitigation of bias has had on the classifications, with a primary focus on false negatives.

Dropping Gender
We find fewer true positives and more false negatives. This is an undesired result. The effect on false positives and true negatives are ambiguous.

Gender Swap
We see more true positives and fewer false negatives, which is desired. However, we also identify more false positives and fewer true negatives.

Disparate Impact Removal
We find fewer true positives and more false negatives. This is undesired.

Learning Fair Representations
We observe fewer false positives and more true negatives, which is desired. However, we see fewer true positives and more false negatives, which is not desired.

Effect on false negatives
Increasing the number of false negatives is an undesired effect of bias mitigation. When assessing the magnitude of the changes in percentage points, we see that the changes in false-negative classifications are highest for LFR, followed by DI-removal. Furthermore, the percentage point changes are smallest for dropping gender, while the changes are acceptable for gender swap, in that the number of false negatives decreases.

In this light, LFR achieves unbiased classifications in the most problematic way. We find a change in percentage points of 1.1% when comparing the false negative classifications of XGBoost trained on LFR data to XGBoost trained on original data (see table 29). To put the effect into perspective, this would mean an increase in false negative classifications of approximately 29 citizens if the LFR-version of XGBoost were to classify all 2600 citizens currently under the care of the Referral Unit of Aalborg Municipality. We have chosen to comment on the results for XGBoost since Aalborg Municipality is expected to use this algorithm.

11.6 Sub-conclusion: Mitigation of bias in AIR

Based on section 11, we have answered research question 2. We have tested techniques for mitigation of bias in the AIR project in the following way:

Dropping gender
In section 11.1, we attempted to mitigate bias by dropping the protected variable, gender, from the data set. From the results, we conclude that the technique successfully mitigated the identified bias. Furthermore, mitigation entailed undesired but small changes in the classifications and predicted probabilities.

Gender swapping
In section 11.2, we attempted to mitigate bias by creating a gender swapped version of the data set. From the results, we conclude that the technique successfully mitigated the identified bias. Furthermore, the changes in classifications were not problematic, and the changes in predicted probabilities were as desired.

Disparate impact removal
In section 11.3, we attempted to mitigate bias by running the disparate impact removal algorithm on the numerical features of the AIR data set. From the results, we conclude that the technique successfully mitigated the identified bias. However, the changes in classifications and predicted probabilities were somewhat problematic.

Learning fair representations
In section 11.4, we attempted to mitigate bias by learning a fair representation of the data set. From the results, we conclude that the technique successfully mitigated the identified bias. However, the changes in classifications and predicted probabilities were somewhat problematic.

12 Discussion

In this section, we critically reflect on specific elements of our thesis that could have been done differently and discuss alternative routes of action and what impact they would have had on the conclusions of the thesis. We structure the discussion into three sections. In the first section we discuss challenges regarding the mitigation techniques, in the second section we criticize technical aspects and our modelling choices, and in the third section we reflect on our theoretical and methodological approach.

12.1 Challenges regarding the mitigation techniques

12.1.1 Dropping gender

The approach of dropping gender can be problematic since other features can be correlated with the protected variable. These features can act as *proxies* for gender, which would enable the model to be biased even though the variable has been dropped [15][19]. The problem of proxies is related to what Calmon et al. call *indirect discrimination*. We do not explore the data set to see if any of the variables could act as potential proxies for gender. For example, one could imagine that some specific aids are only given to either women or men.

12.1.2 Gender Swap

Gender swap might only work when the subgroups of interest do not come from very dissimilar distributions. One could imagine, that the performance of a classifier could be negatively impacted, if it were trained on a data set with "swapped" observations, that are very far from the true distribution of the subgroup. This could worsen the performance of the classifier, since the model could extrapolate in a way that might not be correct. This could particularly be the case, if the mapping from an observed variable to an outcome variable runs through an unobserved characteristic that is unevenly distributed between the subgroups. As an example, consider the modelling of *hours in the gym* (x) on *absolute change in muscle mass* (y), where the mapping between x and y might go through the unobserved *level of testosterone* (z), which would be unevenly distributed between females and males. Using gender swap in this example, could possibly lead to underestimating the change in muscle mass for males and overestimating the change in muscle mass for females.

12.1.3 Disparate impact removal

In the AIF360 library, we have noticed that the algorithm used in the disparate impact implementation does not find a median distribution as described in the literature review (section 5). Instead, it moves the values of the group with the numerically highest distribution towards the distribution of the group with the lowest values. For example, since females are older than males in the AIR data set, the distribution of age for females is moved to resemble the distribution of age for males. We have not attempted to implement a version where the median distribution is found. Removing disparate impact using the median distribution could perhaps have changed the results of the thesis.

12.1.4 Learning fair representations

In the AIF360 implementation of learning fair representations (LFR), several hyperparameters can be tuned - such as the number of prototype vectors representing the original data. In section 11, results showed that the mitigation technique leads to problematic changes in the classifications and predicted probabilities. The results could potentially have been different if the hyperparameters were changed. For example, changing the fairness constraint term weight, A_z, in equation (19), could have enforced a more dissimilar mapping of individuals in/not in the protected group. Furthermore, the mitigation technique is based on several mathematical data transformations. Since the AIR model is intended to be used as a decision support tool in the public sector, it could be challenging to explain LFR in a non-technical way.

12.2 Technical aspects and modelling choices

12.2.1 Choice of 95% as the level for confidence intervals

When identifying bias between genders, we estimate the gender-specific classification rates of a given model and calculate the 95% confidence intervals. If the confidence intervals overlap, we deem it improb-

able that the estimates are different from one another. This comparison of 95% confidence intervals can be seen as a hypothesis test using an 0.05 alpha level. Interpreted as such, our method would test the null hypothesis that an estimate of a given classification rate is the same for females and males. If the 95% confidence intervals do not overlap, we reject the null hypothesis and state that we are 95% certain that the actual classification rate for females and males are different from each other.

We compare the confidence intervals of men and women across 4 classification rates x 5 models x 5 states (4 mitigation techniques and 1 original), which gives 100 comparisons or, if interpreted as above, 100 hypothesis tests. This is quite a large number. When performing multiple hypothesis tests, the risk of false discovery is substantial [40, p. 686]. In other words, rejecting the null hypothesis would be to conclude that the true level of a classification rate is different between genders even though they are the same. We calculate the family-wise error rate (FWER) by: $FWER \leq 1 - (1 - \alpha)^M$, where M is the number of hypotheses and α is the alpha-level [40, p. 686]. Using $M = 100$ and $\alpha = 0.05$, one can see that we have a very high probability of at least one type 1 error:

$$FWER \leq 1 - (1 - 0.05)^{100} = 0.994 \tag{39}$$

We deliberately do not comment on whether or not the estimates are significantly different from one another. However, our method is still at risk of falsely concluding that there could be issues with bias from the fact that the confidence intervals do not overlap in the same way as would have been the case if we did multiple hypothesis tests with 0.05 alpha levels. This being said, we do find consistent results across all five models (SVM, LR, RF, FFNN, and XGBoost), where we find confidence intervals which do not overlap for the classification rates estimated on the original data, while all four mitigation techniques result in mostly overlapping confidence intervals for the relevant classification rates. Furthermore, many of our comparisons of classification rates are correlated, in that we assess the difference between females and males on the same classification rates generated by different models on augmented data sets. Finally, the problem is limited to type 1 errors, which would be to falsely conclude that the classification rates are different, which could lead to a false discovery of bias. However, most of the 100 comparisons of classification rates are made on results after mitigating, where we typically conclude that bias cannot be identified.

We could have used a more strict confidence interval (for example, 99.9% confidence intervals), which would have reduced the probability of type 1 errors. This strategy would have the same effect as, for example, Bonferroni correction. However, this would increase the probability of type 2 errors, in our case, incorrectly concluding that bias cannot be identified. This is perhaps a more undesired type of error for our thesis, since it could leads us to falsely conclude that: 1) bias cannot be identified in the original models and 2) a mitigation technique effectively removed bias.

12.2.2 Robustness of the metric relations

For evaluating the difference in the classification rates, the relation between females' and males' metrics are compared. The metrics for the models built on the original data are plotted in section 10.2, table 22. The metrics are based on binary predictions, where citizens are predicted to be either falling or not. The relations of the metrics could potentially differ if the thresholds of the binary classifier are changed, which might lead to different conclusion regarding the effect of the mitigation techniques. Therefore, we assess the robustness of the relations by testing out different thresholds for the binary classifications of the original models. Plots can be found in appendix, section B.

As a robustness check, we have evaluated the relations in the original models when using a classification threshold between 0.4 and 0.6 to see how it could change the conclusion of the identified gender bias. If the threshold is above 0.5, the TPR relation of the LR moves outside the 80%, in favor of males. Furthermore, if the threshold is above 0.5, the FPR relation of the LR and XGBoost moves below the lower bound, meaning that the classifier is advantageous for males. Finally, if the thresholds move toward 0.4, the FNR relation for XGBoost moves above the upper bound of the 80% region. This means that the relation becomes disadvantageous for females. The robustness check gives the indication that there is some instability for the above mentioned relations. However, when assessing the relation for the FPR and FNR with respect to the binary classification threshold, the relation between the genders are still biased as concluded in the bias mitigation section.

12.2.3 Model performance

For assessing and comparing model performance when testing mitigation techniques, we use the accuracy of the models as e.g. shown in table 23. As shown in table 4, around 78% of the citizens in the data set have not fallen. A naive classifier that only predicts the dominant "No fall" class, would reach an accuracy of 78%. In this unbalanced setting, it can be difficult to interpret the actual performance of the model from the accuracy, since the high performance on the actual negative class might overshadow problems regarding lower performance the actual positive class. Instead of using accuracy, we could have used the receiver operating characteristic curves (ROC curves) to assess model performance. Since ROC uses the TPR and FPR, we would have avoided the issues mentioned above. In appendix C we have provided ROC curves including the area under curve (AUC). When assessing the changes in AUC for the different mitigation strategies compared to the original models, we find moderate drops in AUC of at most 3%, which is very much in line with the changes found in accuracy. Because of this, our conclusions in the analysis would not have been impacted by using AUC instead of accuracy. However, a disadvantage of AUC is that false negatives are not a part of the metric, and as mentioned frequently, this is the most costly misclassification in the AIR context.

12.2.4 Choosing features for mitigation - feature importance

Disparate impact removal and learning fair representations are only applied to the numerical features of the data set. This is somewhat problematic since the numerical features might not affect the model output in a way that could enforce gender bias, and since only 3 out of 136 variables are numerical, while the remaining are one-hot-encoded. The initial descriptive analysis in section 9 only gives an overall understanding of the numerical features' distributions, but the thesis does not examine how the features in each of the five models affect whether a citizen is classified as "Fall" or "Not Fall". However, if the numerical variables impact the classifications of the models to a high degree, it can be argued that only changing the values of 3 out of 136 variables could still have an impact on the classifications of a model. To assess the feature importance of the original models, we have produced plots of the five models' SHAP values in the appendix, section D. SHAP is used for explaining how much each feature contributes to the prediction [56]. In table 47, the top 15 features with the highest absolute mean SHAP value are plotted. For four of the models, the numerical features *Number of aids*, *Loan period* and *Age* are among the top five features with highest mean SHAP values. The features' contributions to the output can both be positive or negative, which is visualized in figure 48 and 49. Overall, it can be seen that the models do not show a clear picture of how the numerical features contribute to the output in terms of positive and negative contributions. Based on the SHAP values, the numerical features chosen for bias mitigation are among the features which contribute the most to the output when assessing the absolute SHAP. This may be an argument as to why the numerical values make sense to use for bias mitigation in disparate impact removal and learning fair representations, even though we only transform 3 out of 136 features. We could have assessed the feature importance before implementing the disparate impact and learning fair representations techniques, but we considered it to be out of the scope of the analysis.

12.2.5 SVM probabilities

For identifying bias in the AIR models, we assessed the predicted probabilities of the models. The SVM is different than the rest of the models, in that it does not use probabilities when classifying. The SVM finds an optimal separating hyperplane for classifying if a citizen falls or not. Therefore, binary classification is not based on a probability threshold. The probability distribution of the SVM output lies approximately between 0.0 and 0.7. One should take this difference between the SVM and the rest of the models into account when comparing the probability outputs of the SVM to the rest of the models. Since the probabilities from an SVM model are somewhat unreliable, we would not recommend that the AIR project team use the SVM probabilities to calculate a risk score.

12.2.6 Tuning of hyperparameters

During the development of the five models, several hyperparameters are chosen for the models. According to Hellström et al. [15], bias can occur when hyperparameters are manually set. A critique of the model development in this thesis is that we have not tried different parameter settings for the models and not tuned using cross validation, for example, to see which parameters most effectively mitigate bias. This could have been explored. However, we did test a few alternative hyperparameter settings for

some of the models during development but did not find any substantial changes in classification rates. Finally, we chose to focus on testing mitigation strategies rather than changing the hyperparameters of the algorithms. Furthermore, since the data set is unbalanced with respect to the output variable (Fall), we could have used stratified cross validation to test if it could improve the models.

12.3 Theoretical and methodological approach

12.3.1 Unobserved sources of bias

This section briefly explores some unobserved sources of bias that we do not take into account when identifying and mitigating bias and how these might introduce bias into the AIR project context. To this end, we use two theoretical concepts from Suresh & Guttag [57]: *measurement bias* and *representation bias*.

Measurement bias and representation bias
The concept of *measurement bias* considers group-dependent differences in choosing, collecting, and computing the variables that are to be used in the machine learning models [57]. Suresh & Guttag argue that there can be group-dependent differences in the quality of data, which can lead to biased outcomes. For example, one could image group dependent differences between females and males when it comes to self-reporting fall incidents. Suresh & Guttag state that *"women are more likely to be misdiagnosed or not diagnosed for conditions where self-reported pain is a symptom"* [57], in the sense that *"professionals hold stereotypic views of women as emotionally labile and more apt to exaggerate complaints of pain than men"* [58]. Suppose a similar dynamic is present in the AIR case, where physiotherapists and caseworkers hold this stereotypical view of women. In that case, it might result in group-dependent noise, where the fall incidents of women are structurally under-registered. Our thesis does not attempt to gather data regarding this potential bias, nor take it into account during our analysis.

The concept of *representation bias* covers the fact that the sampling method might only reach a portion of the population [57]. For example, one could imagine that a higher proportion of the most resourceful citizens in Aalborg Municipality might use private alternatives to public health care, while the least resourceful to a lower degree contact the municipality in due time to receive the aids they need when they need them. If this is the case, then both the fall prediction model and the bias mitigation techniques will potentially perform worse for the most resourceful and the least resourceful citizens in the population since they are under-represented in the data set.

By not exploring these two avenues of bias, we could risk recommending a course of action for bias mitigation that does not solve the issue. For example, if fall incidents for females are under-registered, a more sensible course of action would be to put effort into rectifying this difference between men and women rather than attempting to mitigate a bias that is identified on incorrect data.

Creating new bias?
Finally, by changing the predictions of the algorithm with a bias mitigation technique, hereby possibly creating more false negatives for males, we insert a new source of bias to the situation. Maybe the men, who are now incorrectly classified, have some characteristic that we now are creating new bias against. This could have been explored.

12.3.2 Social context of the algorithm

Our approach to bias identification and mitigation is exclusively data-oriented. However, we could have supported this approach by gathering qualitative empirical evidence, which would have given us an understanding of the social context of the AIR algorithm. By not doing so we step into two "traps" highlighted by Selbst et al., namely the *formalism trap* and the *ripple effect trap* [59].

The *formalism trap* states that machine learning projects run the risk of failing to account for the entire meaning of fairness by reducing it to mathematical formulations [59]. By reducing the identification of bias to non-overlapping confidence intervals and relations outside the 80% region for classification rates, we might over-simplify bias in the AIR context. One could easily imagine that the 80% region is either too relaxed or too strict when evaluating bias in the AIR context. Furthermore, it could be the case that our focus on the overall classification rates is a too broad notion of bias or that we should have chosen

a different way to identify bias altogether. A more detailed investigation of the view on bias could have been performed by interviewing stakeholders of the AIR project. This could potentially give a more valid foundation for defining bias relation thresholds.

The *ripple effect trap* states that some machine learning projects fail to understand how the implementation of technology might impact a pre-existing system, hereby overlooking possible unintended consequences. By not gathering and analysing qualitative data, we fail to fully understand the social and organizational context of the AIR algorithm. Therefore, we run the risk of recommending bias mitigation techniques that are unfit for implementation in the AIR project because of an unintended consequence we did not consider.

12.3.3 Should we even be doing this?

We end the discussion on an ethical note, considering a question that Rachel Thomas from San Francisco University recommends practitioners to ask themselves at the beginning of an AI project, which is: *"should we even be doing this?"* [60]. She presents two examples where her question is particularly relevant: an algorithm that uses facial recognition to classify faces as Chinese Uyghur, Tibetan, and Korean and another facial recognition algorithm that classifies faces as homosexual or heterosexual. In the wrong hands, it is easy to see how these algorithms could be used to oppress certain vulnerable groups. Clearly, the AIR project is very far from being problematic in the same sense as the two examples. However, it is still valuable to consider if one should use machine learning algorithms to predict the fall probability for citizens in Aalborg Municipality. When the algorithm is implemented, it is anticipated that it will be used to support caseworkers in finding additional citizens that would benefit from fall prevention training, hereby providing more welfare to the citizens of Aalborg. However, it cannot be ruled out that the algorithm could be used for different purposes. If the algorithm were to be used in a scenario with more limited resources and therefore should support caseworkers in deciding who does and who does not have a high enough probability of falling to be allocated training, the issues with bias in the algorithm would be more severe and one could argue that using an algorithmic approach is inappropriate.

However, simply refraining from training an algorithm or creating bias mitigation techniques will not solve the issues with bias. Humans make biased decisions every day, even without the assistance of algorithms [57] [15]. These biases are, for the most part, implicit and difficult to measure. In contrast, when algorithms are used for decision support, one can attempt to formalize and make explicit the biases of the models and attempt to mitigate them. If the goal is to move towards less biased decisions, one could argue that even insufficient attempts to make bias explicit and mitigate it is a better option than only relying on human judgment.

13 Conclusion

The purpose of the thesis was to identify and mitigate bias in machine learning models built on the data set used in the AIR project. We present the results from the analysis, and use them to provide the AIR project team with actionable recommendations regarding bias identification and mitigation. The research questions are:

- *RQ1: How can bias be identified in the AIR project?*
- *RQ2: How can bias be mitigated in the AIR project?*

13.1 Results

Research question 1 is answered using a two-step strategy. First, we observed that the estimates of differences in classification rates (TPR, FPR, TNR, and FNR) between females and males have non-overlapping confidence intervals for all five models. Second, the relation plot in figure 22 shows that the differences between females and males in terms of FPR and FNR are problematic. This means that we identify bias in the AIR project.

Furthermore, the bias is disadvantageous for females in the sense that males who do not fall are more likely to be provided fall prevention training than females who do not fall (FPR), while females who fall are less likely to be provided fall prevention training than males who fall (FNR). We emphasize that the FNR bias is particularly problematic since falling can have severe consequences for elderly citizens.

Research question 2 is answered by showing that all four bias mitigation techniques (*dropping gender*, *gender swap*, *disparate impact removal* and *learning fair representations*) successfully mitigated bias. However, the techniques did exhibit differences regarding the effect on the classifiers' predicted probabilities, false negative and false positive classifications:

- Dropping gender had undesired but small changes in the classifications and predicted probabilities.
- Gender swapping exhibited changes in the classifications and predicted probabilities that were as desired.
- Disparate impact removal entailed changes in the classifications and predicted probabilities that were somewhat problematic.
- Learning fair representations resulted in problematic changes in classifications and predicted probabilities.

13.2 Recommendations

Short-term recommendations
We recommend that the AIR project team use the thesis's bias identification method when building the AIR classification model. If the thesis's identification method is applied, we recommend that the project team examine whether the 80% is applicable in the AIR context. Furthermore, the team should investigate if other variables than gender are considered protected in the organizational setting. Moreover, the AIR project team could evaluate whether the 95% confidence intervals for estimates of classification rates are appropriate when considering the balance between type 1 and type 2 errors.

We recommend that the AIR project team consider using the two mitigation techniques: Dropping gender and gender swapping.

Of the four tested mitigation techniques, *dropping gender* and *gender swapping* had the most desirable results in terms of classifications and probabilities. Furthermore, the techniques are simple and easy to explain, which we deem advantageous for the AIR project. Gender may seem to be an irrelevant feature to use when predicting the probability of falling. Therefore, dropping gender from the data set is an obvious and easily explained avenue for bias mitigation. However, we also recommend that the team investigates if any features act as proxies for gender since the gender-specific bias could run through the proxies instead. Gender swapping affected the classifications and probabilities in the most desired way.

However, the technique could potentially lead to a decrease in performance if the protected subgroups come from very different distributions.

Nevertheless, the two techniques successfully mitigated the identified gender bias in the AIR project.

Long-term recommendations
In section 8 we describe a two-step system for bias mitigation, where one party attempts to mitigate bias while another party attempts to maximize their utility when building a classifier. If the AIR project team and Aalborg Municipality wish to work with bias mitigation in the future we recommend this approach. By structuring the efforts to mitigate bias in this way, knowledge and practical know-how regarding different bias pre-processing techniques can be collected in one place. This could make it easier to cooperate with other units in Aalborg Municipality and align bias mitigation efforts across the entire organization. The two-step system would enable Aalborg Municipality to cooperate with external organizations or private companies, which might have proprietary algorithms. One could imagine that Aalborg Municipality would be the owners of the data sets that are to be used to build models related to future efforts of bias mitigation. Therefore, it would be sensible to focus on pre-processing mitigation techniques since the municipality could be expected to be responsible for processing and collecting the data and naturally spend resources on pre-processing activities. Furthermore, there could be possibilities in pre-processing mitigation techniques that work across application fields, for example, by mitigating bias in personal data for all citizens in Aalborg.

When focusing on the AIR project, we consider the 80% region to be an appropriate guideline for bias identification. However, other cases might entail that the 80% region is not acceptable as a guiding threshold. For example, suppose Aalborg Municipality was to build an AI to support decisions regarding more consequential interventions in people's lives. In that case, decisions related to the work of social services, one could argue that the acceptable region should be narrower, for instance, 90% or 95%. It depends on the specific scenario, but we recommend that the 80% region is revised and reconsidered for every new area where bias identification and mitigation efforts are made.

As a final remark, we believe that the identification and mitigation techniques presented in the thesis could be useful for future efforts to build less biased machine learning models.

References

[1] Finansministeriet and Erhvervsministeriet, "National strategi for kunstig intelligens," 2019, available at https://www.regeringen.dk/media/6537/ai-strategi_web.pdf, retrieved 09.03.2021.

[2] C. F. Smed. Interview with project owner. Conducted 08.02.2021.

[3] J. Zhao, T. Wang, M. Yatskar, R. Cotterell, V. Ordonez, and K.-W. Chang, "Gender bias in contextualized word embeddings," 2019, available at https://arxiv.org/abs/1904.03310, retrieved 10.02.2021.

[4] L. Sweeney, "Discrimination in online ad delivery," 2013, available at https://arxiv.org/pdf/1301.6822.pdf, retrieved 18.06.2021.

[5] J. Buolamwini and T. Gebru, "Gender Shades results," 2018, available at http://gendershades.org/overview.html, retrieved 11.05.2021.

[6] K. Hao. (2019) Ai is sending people to jail—and getting it wrong. Available at https://www.technologyreview.com/2019/01/21/137783/algorithms-criminal-justice-ai/, retrieved 10.02.2021.

[7] J. Angwin, S. Mattu, J. Larson, and L. Kirchner, "Machine bias: there's software used across the country to predict future crimi- nals. and it's biased against blacks," 2016, available at https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing, retrieved 10.02.2021.

[8] M. Rovatsos, B. Mittelstadt, and A. Koene, "Landscape summary: Bias in algorithmic decision-making," 2019, available at https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/819055/Landscape_Summary_-_Bias_in_Algorithmic_Decision-Making.pdf, retrieved 08.03.2021.

[9] G. Shobha and S. Rangaswamy, "Computational analysis and understanding of natural languages: Principles, methods and applications," ser. Handbook of Statistics, V. N. Gudivada and C. Rao, Eds. Elsevier, 2018, vol. 38, ch. Chapter 8 - Machine Learning, pp. 197–228, available at https://www.sciencedirect.com/science/article/pii/S0169716118300191, retrieved 02.05.2021.

[10] A. Joshi, *Machine learning and artificial intelligence*. Springer, 2020, ch. Chapter 1: Introduction to AI and ML, available at https://doi-org.proxy.findit.dtu.dk/10.1007/978-3-030-26622-6, retrieved 02.05.2021.

[11] S. Verma and J. Rubin, "Fairness definitions explained," 2018, pp. 1–7, available at https://www.ece.ubc.ca/~mjulia/publications/Fairness_Definitions_Explained_2018.pdf, retrieved 02.05.2021.

[12] T. Fawcett, "An introduction to roc analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006, available at https://www.sciencedirect.com/science/article/pii/S016786550500303X, retrieved 02.05.2021.

[13] D. Powers, "Evaluation: From precision, recall and f-factor to roc, informedness, markedness correlation," 2008, available at https://www.researchgate.net/publication/228529307_Evaluation_From_Precision_Recall_and_F-Factor_to_ROC_Informedness_Markedness_Correlation, retrieved 25.06.2021.

[14] E. M. Bender and B. Friedman, "Data statements for natural language processing: Toward mitigating system bias and enabling better science," *Transactions of the Association for Computational Linguistics*, vol. 6, 2018, available at https://www.aclweb.org/anthology/Q18-1041, retrieved 15.02.2021.

[15] T. Hellström, V. Dignum, and S. Bensch, "Bias in machine learning – what is it good for?" 2020, available at https://arxiv.org/abs/2004.00686, retrieved 22.02.2021.

[16] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel, "Fairness through awareness," 2011, available at https://arxiv.org/abs/1104.3913, retrieved 21.02.2021.

[17] M. B. Zafar, I. Valera, M. Gomez Rodriguez, and K. P. Gummadi, "Fairness beyond disparate treatment & disparate impact," 2017, available at http://dx.doi.org/10.1145/3038912.3052660, retrieved 21.02.2021.

[18] B. Jann, "The blinder–oaxaca decomposition for linear regression models," 2008, available at https://doi.org/10.1177/1536867X0800800401, retrieved 03.03.2021.

[19] F. Calmon, D. Wei, B. Vinzamuri, K. Natesan Ramamurthy, and K. R. Varshney, "Optimized pre-processing for discrimination prevention," in *Advances in Neural Information Processing Systems*, 2017, available at https://proceedings.neurips.cc/paper/2017/file/9a49a25d845a483fae4be7e341368e36-Paper.pdf, retrieved 03.03.2021.

[20] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork, "Learning fair representations," 2013, available at http://proceedings.mlr.press/v28/zemel13.html, retrieved 16.03.2021.

[21] M. Feldman, S. Friedler, J. Moeller, C. Scheidegger, and S. Venkatasubramanian, "Certifying and removing disparate impact," 2015, available at https://arxiv.org/pdf/1412.3756.pdf, retrieved 09.05.2021.

[22] California-State-Personnel-Board, "Summary of the uniform guidelines on employee selection procedures," 2003, available at https://spb.ca.gov/content/laws/selection_manual_appendixd.pdf , retrieved 19.06.2021.

[23] M. Hardt, E. Price, and N. Srebro, "Equality of opportunity in supervised learning," 2016, available at https://arxiv.org/abs/1610.02413, retrieved 09.03.2021.

[24] T. Bolukbasi, K.-W. Chang, J. Zou, V. Saligrama, and A. Kalai, "Man is to computer programmer as woman is to homemaker? debiasing word embeddings," 2016, available at https://arxiv.org/abs/1607.06520, retrieved 10.02.2021.

[25] Aarhus.Universitet. (2020) AIR ai rehabilitering. Available at https://projekter.au.dk/air/, retrieved 10.02.2021.

[26] Agency.for.digitisation. (2020) Digitisation on the agenda in the annual budget agreements. Available at https://en.digst.dk/news/news-archive/2020/august/digitisation-on-the-agenda-in-the-annual-budget-agreements/, retrieved 12.02.2021.

[27] Aalborg.Kommune. Myndighedsafdelingen. Available at https://www.aalborg.dk/om-kommunen/organisation/%C3%A6ldre-og-handicapforvaltningen/myndighedsafdelingen, retrieved 12.02.2021.

[28] T. M. of Social Affairs and the Interior. Consolidation act on social services. Available at http://english.sm.dk/media/14900/consolidation-act-on-social-services.pdf, retrieved 12.02.2021.

[29] We have held several meetings with Christian Marius Lillelund regarding development of the AIR machine learning model. The meetings have not been recorded. Read the method section for further details.

[30] Retsinformation. Bekendtgørelse af forvaltningsloven. Available at https://www.retsinformation.dk/eli/lta/2014/433, retrieved 23.06.2021.

[31] C. M. Lillelund, *Internal AIR report*, retrieved 24.05.2021.

[32] Gyldendal. Matematisk modellering. Available at https://kolorit.gyldendal.dk/indhold/kolorit/laerer/pdf/K9%20LV%20light%20Matematisk%20modellering.pdf, retrieved 04.06.2021.

[33] J. C. L. Nielsen, K. Holleufer, and H. H. Bülow, *MetodeNU - introduktion til samfundsfaglige metoder*, 2021, ch. Chapter 2.2: Kvantitativ metode, available at https://metodenu.systime.dk/?id=131, retrieved 04.06.2021.

[34] A. Universitet. Semistruktureret interview. Available at https://metodeguiden.au.dk/semistruktureret-interview/, retrieved 04.06.2021.

[35] ——. Kvalitativ metode. Available at https://metodeguiden.au.dk/kvalitativ-metode/, retrieved 04.06.2021.

[36] J. Larson, S. Mattu, L. Kirchner, and J. Angwin, "How we analyzed the compas recidivism algorithm," 2016, available at https://www.propublica.org/article/how-we-analyzed-the-compas-recidivism-algorithm, retrieved 05.04.2021.

[37] T. A. F. . A. Authors. aif360.algorithms.preprocessing.disparateimpactremover. Available at https://aif360.readthedocs.io/en/latest/modules/generated/aif360.algorithms.preprocessing.DisparateImpactRemover.html, retrieved 11.06.2021.

[38] ——. aif360.algorithms.preprocessing.lfr. Available at https://aif360.readthedocs.io/en/latest/modules/generated/aif360.algorithms.preprocessing.LFR.html, retrieved 22.06.2021.

[39] J. Brownlee, "A tour of machine learning algorithms," 2019, available at https://machinelearningmastery.com/a-tour-of-machine-learning-algorithms/, retrieved 03.05.2021.

[40] T. Hastie, R. Tibshirani, and J. Friedman, "The elements of statistical learning," 2008.

[41] scikit learn, "sklearn. linear model. logisticregression," 2021, available at https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html, retrieved 07.06.2021.

[42] ——, "sklearn.svm.svc," 2021, available at https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html, retrieved 07.06.2021.

[43] ——, "sklearn.ensemble.randomforestclassifier," 2021, available at https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html, retrieved 07.06.2021.

[44] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, ch. Chapter 6: Deep Feedforward Networks, available at https://www.deeplearningbook.org/, retrieved 01.05.2021.

[45] M. A. Nielsen, *Neural networks and deep learning*. Determination press, 2015, ch. Chapter 1: Using neural nets to recognize handwritten digits, Chapter 2: How the backpropagation algorithm works, available at http://neuralnetworksanddeeplearning.com/chap1.html, retrieved 01.05.2021.

[46] xgboost developers. About xgboost. Available at https://xgboost.ai/about, retrieved 24.05.2021.

[47] B. Boehmke and B. Greenwell, *Hands-On Machine Learning with R*, 2020, ch. Chapter 12: Gradient Boosting, available at https://bradleyboehmke.github.io/HOML/gbm.html, retrieved 25.05.2021.

[48] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," 2016, available at http://arxiv.org/abs/1603.02754, retrieved 24.06.2021.

[49] Y. Jin. Tree boosting with xgboost – why does xgboost win "every" machine learning competition? Available at https://syncedreview.com/2017/10/22/tree-boosting-with-xgboost-why-does-xgboost-win-every-machine-learning-competition/, retrieved 24.06.2021.

[50] xgboost developers. Introduction to boosted trees. Available at https://xgboost.readthedocs.io/en/latest/tutorials/model.html, retrieved 24.06.2021.

[51] J. Brownlee. Tune xgboost performance with learning curves. Available at https://machinelearningmastery.com/tune-xgboost-performance-with-learning-curves/, retrieved 24.06.2021.

[52] C. F. Pedersen. Air (gitlab). Available at https://gitlab.au.dk/cfp/air, retrieved 24.06.2021.

[53] scikit learn, "1.4 support vector machines," 2021, available at https://scikit-learn.org/stable/modules/svm.html#id11, retrieved 07.06.2021.

[54] J. C. Platt. Probabilistic outputs for svms and comparisons to regularized likelihood methods. Available at https://home.cs.colorado.edu/~mozer/Teaching/syllabi/6622/papers/Platt1999.pdf, retrieved 07.06.2021.

[55] D. Statistik. Middellevetiden stiger fortsat. Available at https://www.dst.dk/da/Statistik/nyt/NytHtml?cid=30217, retrieved 07.06.2021.

[56] C. Molnar, *Interpretable Machine Learning*, 2019, ch. Chapter 5.9-5.10, available at https://christophm.github.io/interpretable-ml-book/, retrieved 04.06.2021.

[57] H. Suresh and J. V. Guttag, "A framework for understanding unintended consequences of machine learning," 2020, available at https://arxiv.org/abs/1901.10002, retrieved 10.02.2021.

[58] K. L. Calderon, "The influence of gender on the frequency of pain and sedative medication administered to postoperative patients," 1990, available at https://link.springer.com/article/10.1007/BF00289259 , retrieved 18.06.2021.

[59] A. Selbst, D. Boyd, S. Friedler, S. Venkatasubramanian, and J. Vertesi, "Fairness and abstraction in sociotechnical systems," pp. 59–68, 01 2019, available at https://dl.acm.org/doi/10.1145/3287560.3287598, retrieved 01.03.2021.

[60] R. Thomas, "Getting specific about algorithmic bias - rachel thomas," 2019, available at https://www.youtube.com/watch?v=S-6YGPrmtYc&t=152s, retrieved 18.06.2021.

[61] K. Crawford, "The trouble with bias - nips 2017 keynote - kate crawford," 2017, youtube, Available at https://www.youtube.com/watch?v=fMym_BKWQzk, retrieved 16.03.2021.

[62] N. Lee, P. Resnick, and G. Barton, "Algorithmic bias detection and mitigation: Best practices and policies to reduce consumer harms," 2019, available at https://www.brookings.edu/research/algorithmic-bias-detection-and-mitigation-best-practices-and-policies-to-reduce-consumer-harms/, retrieved 01.03.2021.

[63] K. Kirkpatrick, "It's not the algorithm, it's the data," 2017, available at https://doi.org/10.1145/3022181, retrieved 06.04.2021.

[64] "compas-analysis," 2016, available at https://raw.githubusercontent.com/propublica/compas-analysis/master/compas-scores-two-years.csv, retrieved 06.04.2021.

[65] FindLaw.com, "What's the difference between a misdemeanor vs. felony?" 2019, available at https://www.findlaw.com/criminal/criminal-law-basics/what-distinguishes-a-misdemeanor-from-a-felony.html, retrieved 13.04.2021.

A Bias metrics for the five models built on AIR data set

Gender	TPR	FPR	TNR	FNR
Support Vector Machine				
Female	**63.6**	**25.7**	**74.3**	**36.4**
	(61.8-65.3)	(24.9-26.6)	(73.4-75.1)	(34.7-38.2)
Male	**78.0**	**38.2**	**61.8**	**22.0**
	(75.4-80.7)	(36.8-39.6)	(60.4-63.2)	(19.3-24.6)
Logistic Regression				
Female	**53.9**	**33.8**	**66.2**	**46.1**
	(52.1-55.7)	(32.9-34.6)	(65.4-67.1)	(44.3-47.9)
Male	**68.7**	**44.4**	**55.6**	**31.3**
	(66.6-70.9)	(43.1-45.7)	(54.3-56.9)	(29.1-33.4)
Random Forest				
Female	**55.0**	**4.5**	**95.5**	**45.0**
	(53.3-56.8)	(4.2-4.9)	(95.1-95.8)	(43.2-46.7)
Male	**63.0**	**5.7**	**94.3**	**37.0**
	(60.6-65.3)	(5.0-6.4)	(93.6-95.0)	(34.7-39.4)
FFNN				
Female	**33.5**	**6.6**	**93.4**	**66.5**
	(31.0-36.1)	(5.8-7.3)	(92.7-94.2)	(63.9-69.0)
Male	**39.0**	**8.8**	**91.2**	**61.0**
	(35.4-42.6)	(7.7-9.9)	(90.1-92.3)	(57.4-64.6)
XGBoost				
Female	**59.8**	**13.7**	**86.3**	**40.2**
	(57.7-62.0)	(12.4-15.0)	(85.0-87.6)	(38.0-42.3)
Male	**68.3**	**17.4**	**82.6**	**31.7**
	(65.9-70.6)	(15.7-19.2)	(80.8-84.3)	(29.4-34.1)

Table 15: Classification metrics of algorithms grouped by gender. The original models.

Gender	Accuracy
Support Vector Machine	
Female:	**72.0** (71.3-72.7)
Male:	**65.7** (64.8-66.7)
Total:	**69.7** (69.2-70.3)
Logistic Regression	
Female:	**63.7** (63.0-64.4)
Male:	**58.9** (57.7-60.0)
Total:	**61.9** (61.4-62.4)
Random Forest	
Female:	**87.1** (86.6-87.5)
Male:	**86.4** (85.7-87.1)
Total:	**86.8** (86.4-87.2)
FFNN	
Female	**81.0** (80.3-81.6)
Male	**78.1** (77.2-79.1)
Total	**79.9** (79.3-80.5)
XGBoost	
Female	**80.5** (79.9-81.1)
Male	**79.5** (78.4-80.5)
Total	**80.1** (79.5-80.7)

Table 16: Accuracy of algorithms grouped by gender. Original models.

Gender	TPR	FPR	TNR	FNR
Support Vector Machine				
Female	**65.4**	**27.2**	**72.8**	**34.6**
	(63.7-67.2)	(26.3-28.1)	(71.9-73.7)	(32.8-36.3)
Male	**68.5**	**30.0**	**70.0**	**31.5**
	(65.7-71.3)	(28.8-31.2)	(68.8-71.2)	(28.7-34.3)
Logistic Regression				
Female	**57.2**	**37.2**	**62.8**	**42.8**
	(55.4-59.0)	(36.3-38.0)	(62.0-63.7)	(41.0-44.6)
Male	**64.1**	**39.4**	**60.6**	**35.9**
	(62.0-66.2)	(38.1-40.7)	(59.3-61.9)	(33.8-38.0)
Random Forest				
Female	**55.6**	**5.0**	**95.0**	**44.4**
	(53.9-57.4)	(4.6-5.4)	(94.6-95.4)	(42.6-46.1)
Male	**59.2**	**4.6**	**95.4**	**40.8**
	(56.8-61.7)	(4.0-5.2)	(94.8-96.0)	(38.3-43.2)
FFNN				
Female	**33.2**	**6.7**	**93.3**	**66.8**
	(30.1-36.3)	(5.9-7.5)	(92.5-94.1)	(63.7-69.9)
Male	**35.1**	**8.0**	**92.0**	**64.9**
	(32.2-38.1)	(6.9-9.0)	(91.0-93.1)	(61.9-67.8)
XGBoost				
Female	**61.5**	**17.0**	**83.0**	**38.5**
	(59.5-63.5)	(15.9-18.1)	(81.9-84.1)	(36.5-40.5)
Male	**66.8**	**16.2**	**83.8**	**33.2**
	(64.3-69.3)	(15.1-17.3)	(82.7-84.9)	(30.7-35.7)

Table 17: Classification metrics of algorithms grouped by gender - Dropping the protected variable

Gender	Accuracy
Support Vector Machine	
Females:	**71.3** (70.6-72.0)
Males:	**69.6** (68.6-70.5)
Total:	**70.7** (70.1-71.2)
Logistic Regression	
Females:	**61.7** (60.9-62.4)
Males:	**61.5** (60.4-62.6)
Total:	**61.6** (61.1-62.1)
Random Forest	
Females:	**86.82** (86.34-87.31)
Males:	**86.29** (85.55-87.02)
Total:	**86.63** (86.2-87.06)
FFNN	
Female	**80.8** (80.2-81.4)
Male	**77.8** (76.9-78.8)
Total	**79.7** (79.2-80.3)
XGBoost	
Female	**78.5** (77.6-79.4)
Male	**79.4** (78.3-80.5)
Total	**78.8** (78.1-79.6)

Table 18: Accuracy of algorithms grouped by gender when dropping gender variable.

Model	Females (Fall)	Males (Fall)	Females (No Fall)	Males (No Fall)
SVM	31.6	32.7	19.4	20.1
	(31.1-32.1)	(32.0-33.3)	(19.2-19.6)	(19.8-20.5)
LR	52.0	54.1	44.0	44.4
	(51.4-52.6)	(53.4-54.8)	(43.7-44.4)	(43.9-44.9)
RF	50.2	53.3	15.0	14.5
	(49.2-51.2)	(52.0-54.5)	(14.7-15.3)	(14.0-14.9)
FFNN	35.5	36.4	18.0	19.0
	(34.7-36.4)	(35.3-37.4)	(17.7-18.3)	(18.6-19.4)
XGBoost	55.8	58.0	23.1	23.9
	(54.8-56.9)	(56.8-59.2)	(22.7-23.6)	(23.3-24.5)

Table 19: Mean and 95% confidence interval for predicted probabilities conditioned by actual fall and gender. From models trained on data where gender is dropped.

Gender	TPR	FPR	TNR	FNR
Support Vector Machine				
Female:	**67.0**	**24.1**	**75.9**	**33.0**
	(65.4-68.6)	(23.3-24.8)	(75.2-0.8)	(31.4-34.6)
Male:	**69.5**	**27.8**	**72.19**	**30.5**
	(67.0-72.1)	(26.8-28.8)	(71.2-73.2)	(27.9-33.0)
Logistic Regression				
Female:	**57.3**	**37.0**	**63.0**	**42.7**
	(55.5-59.2)	(36.2-37.9)	(62.1-0.6)	(40.8-44.5)
Male:	**64.7**	**38.8**	**61.16**	**35.3**
	(62.6-66.8)	(37.5-40.2)	(59.8-62.5)	(33.2-37.4)
Random Forest				
Female:	**57.5**	**5.4**	**94.6**	**42.5**
	(55.8-59.3)	(5.0-5.8)	(94.2-1.0)	(40.7-44.2)
Male:	**60.5**	**4.9**	**95.15**	**39.5**
	(57.9-63.1)	(4.2-5.5)	(94.5-95.8)	(36.9-42.1)
FFNN				
Female	**35.8**	**7.3**	**92.7**	**64.2**
	(33.1-38.6)	(6.5-8.1)	(91.9-93.5)	(61.4-66.9)
Male	**38.6**	**8.5**	**91.5**	**61.4**
	(34.8-42.4)	(7.5-9.5)	(90.5-92.5)	(57.6-65.2)
XGBoost				
Female	**61.2**	**15.3**	**84.7**	**38.8**
	(59.1-63.3)	(14.2-16.5)	(83.5-85.8)	(36.7-40.9)
Male	**65.6**	**14.8**	**85.2**	**34.4**
	(63.5-67.8)	(13.3-16.3)	(83.7-86.7)	(32.2-36.5)

Table 20: Performance metrics of algorithms grouped by gender - Gender Swap

Gender	**Accuracy**
Support Vector Machine	
Female:	**74.1** (73.4-74.8)
Male:	**71.5** (70.6-72.3)
Total:	**73.1** (72.6-73.7)
Logistic Regression	
Female:	**61.8** (61.0-62.5)
Male:	**62.1** (60.9-63.2)
Total:	**61.9** (61.4-62.4)
Random Forest	
Female:	**86.9** (86.4-87.3)
Male:	**86.4** (85.6-87.2)
Total:	**86.7** (86.3-87.1)
FFNN	
Female	**80.8** (80.2-81.4)
Male	**78.3** (77.2-79.3)
Total	**79.9** (79.3-80.5)
XGBoost	
Female	**79.8** (78.9-80.6)
Male	**80.3** (79.2-81.5)
Total	**80.0** (79.1-80.9)

Table 21: Accuracy of algorithms grouped by gender when swapping gender.

Model	Females (Fall)	Males (Fall)	Females (No Fall)	Males (No Fall)
SVM	38.4	39.7	16.9	18.0
	(37.6-39.3)	(38.7-40.7)	(16.6-17.3)	(17.6-18.5)
LR	52.1	54.4	43.6	43.9
	(51.5-52.7)	(53.7-55.1)	(43.3-44.0)	(43.3-44.4)
RF	55.1	57.4	13.7	13.1
	(54.0-56.3)	(56.0-58.8)	(13.3-14.1)	(12.6-13.5)
FFNN	35.1	36.7	17.8	18.6
	(34.2-35.9)	(35.7-37.7)	(17.5-18.0)	(18.2-19.0)
XGBoost	55.9	57.9	21.4	21.8
	(54.7-57.0)	(56.6-59.2)	(21.0-21.9)	(21.2-22.4)

Table 22: Mean and 95% confidence interval for predicted probabilities conditioned by actual fall and gender. From models trained on gender swapped data.

Gender	TPR	FPR	TNR	FNR
Support Vector Machine				
Female:	**64.8**	**27.5**	**72.5**	**35.2**
	(63.0-66.6)	(26.5-28.4)	(71.6-0.7)	(33.4-37.0)
Male:	**68.9**	**29.9**	**70.1**	**31.1**
	(66.4-71.5)	(28.9-30.9)	(69.1-71.1)	(28.5-33.6)
Logistic Regression				
Female:	**57.8**	**37.9**	**62.1**	**42.2**
	(56.1-59.5)	(36.9-38.9)	(61.1-0.6)	(40.5-43.9)
Male:	**63.9**	**38.8**	**61.24**	**36.1**
	(61.8-66.0)	(37.5-40.0)	(60.0-62.5)	(34.0-38.2)
Random Forest				
Female:	**48.0**	**5.0**	**95.0**	**52.0**
	(46.1-49.9)	(4.5-5.4)	(94.6-1.0)	(50.1-53.9)
Male:	**56.1**	**4.6**	**95.39**	**43.9**
	(53.7-58.6)	(4.0-5.2)	(94.8-96.0)	(41.4-46.3)
FFNN				
Female	**32.2**	**6.5**	**93.5**	**67.8**
	(28.9-35.6)	(5.5-7.5)	(92.5-94.5)	(64.4-71.1)
Male	**34.4**	**9.3**	**90.7**	**65.6**
	(30.6-38.2)	(8.0-10.7)	(89.3-92.0)	(61.8-69.4)
XGBoost				
Female	**55.0**	**16.2**	**83.8**	**45.0**
	(52.4-57.5)	(14.7-17.8)	(82.2-85.3)	(42.5-47.6)
Male	**60.5**	**16.5**	**83.5**	**39.5**
	(58.1-62.9)	(14.7-18.2)	(81.8-85.3)	(37.1-41.9)

Table 23: Classification metrics of algorithms grouped by gender - Disparate Impact Removal

Gender	**Accuracy**
Support Vector Machine	
Female:	**70.9** (70.2-71.7)
Male:	**69.8** (69.0-70.6)
Total:	**70.5** (70.0-71.1)
Logistic Regression	
Female:	**61.2** (60.4-62.0)
Male:	**62.0** (60.8-63.1)
Total:	**61.5** (60.9-62.0)
Random Forest	
Female:	**85.2** (84.7-85.8)
Male:	**85.5** (84.8-86.2)
Total:	**85.3** (84.9-85.8)
FFNN	
Female	**80.7** (80.1-81.3)
Male	**76.6** (75.8-77.4)
Total	**79.2** (78.7-79.7
XGBoost	
Female	**77.8** (76.7-78.9)
Male	**77.8** (76.5-79.0)
Total	**77.8** (76.8-78.9)

Table 24: Accuracy of algorithms grouped by gender - Disparate Impact Removal.

Model	Females (Fall)	Males (Fall)	Females (No Fall)	Males (No Fall)
SVM	31.5	33.0	19.4	20.2
	(31.0-32.1)	(32.4-33.7)	(19.1-19.6)	(19.8-20.5)
LR	52.3	54.1	44.4	44.0
	(51.7-52.9)	(53.4-54.8)	(44.0-44.7)	(43.5-44.5)
RF	45.0	49.7	15.8	15.2
	(44.1-45.9)	(48.5-50.9)	(15.5-16.2)	(14.8-15.7)
FFNN	35.4	35.7	18.6	19.0
	(34.6-36.2)	(34.7-36.7)	(18.3-18.9)	(18.6-19.4)
XGBoost	50.6	54.2	23.9	24.0
	(49.6-51.7)	(52.9-55.5)	(23.5-24.4)	(23.4-24.7)

Table 25: Mean and 95% confidence interval for predicted probabilities conditioned by actual fall and gender. From models trained on DI removal data.

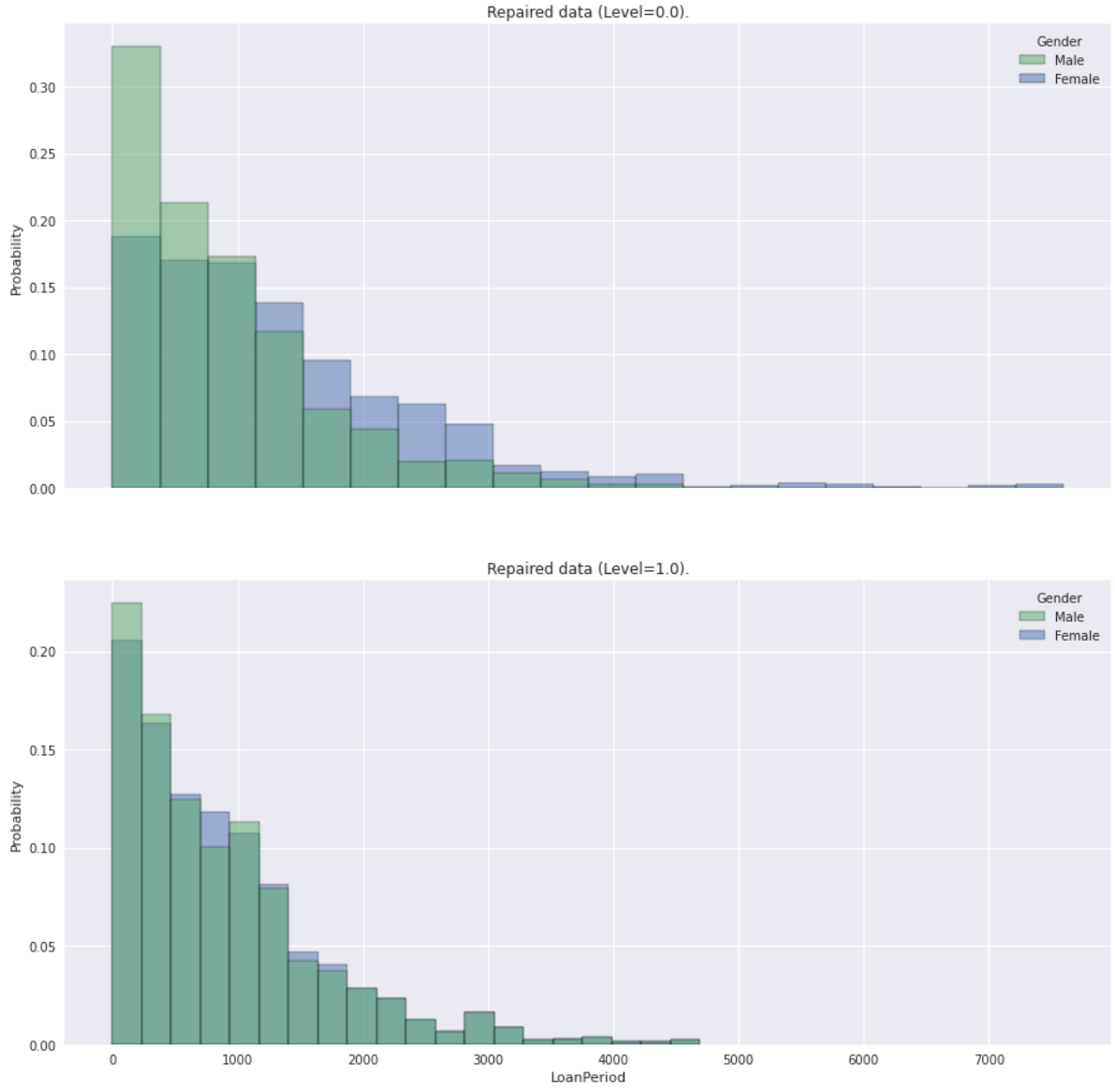


Figure 36: Distributions of LoanPeriod for females (blue) and males (green) using the disparate impact removal method at different repairing levels

Gender	**TPR**	**FPR**	**TNR**	**FNR**
Support Vector Machine				
Female:	**64.4**	**27.5**	**72.5**	**35.6**
	(62.6-66.2)	(26.7-28.2)	(71.8-0.7)	(33.8-37.4)
Male:	**66.1**	**31.7**	**68.25**	**33.9**
	(63.5-68.7)	(30.5-33.0)	(67.0-69.5)	(31.3-36.5)
Logistic Regression				
Female:	**60.5**	**37.9**	**62.1**	**39.5**
	(58.7-62.4)	(37.2-38.7)	(61.3-0.6)	(37.6-41.3)
Male:	**62.0**	**39.0**	**61.05**	**38.0**
	(59.8-64.3)	(37.6-40.3)	(59.7-62.4)	(35.7-40.2)
Random Forest				
Female:	**52.0**	**4.2**	**95.8**	**48.0**
	(50.3-53.7)	(3.9-4.6)	(95.4-1.0)	(46.3-49.7)
Male:	**58.1**	**5.5**	**94.5**	**41.9**
	(55.9-60.2)	(4.9-6.1)	(93.9-95.1)	(39.8-44.1)
FFNN				
Female	**22.9**	**5.6**	**94.4**	**77.1**
	(20.0-25.8)	(4.8-6.3)	(93.7-95.2)	(74.2-80.0)
Male	**24.5**	**6.6**	**93.4**	**75.5**
	(21.1-27.9)	(5.6-7.6)	(92.4-94.4)	(72.1-78.9)
XGBoost				
Female	**55.8**	**14.3**	**85.7**	**44.2**
	(53.6-57.9)	(12.9-15.6)	(84.4-87.1)	(42.1-46.4)
Male	**63.7**	**16.2**	**83.8**	**36.3**
	(61.2-66.2)	(14.8-17.6)	(82.4-85.2)	(33.8-38.8)

Table 26: Classification metrics of algorithms grouped by gender - Learning Fair Representations

Gender	Accuracy
Support Vector Machine	
Female:	**70.8** (70.2-71.4)
Male:	**67.7** (66.8-68.5)
Total:	**69.7** (69.2-70.1)
Logistic Regression	
Female:	**61.7** (61.0-62.4)
Male:	**61.4** (60.2-62.5)
Total:	**61.6** (61.1-62.1)
Random Forest	
Female:	**86.7** (86.2-87.1)
Male:	**85.3** (84.6-86.0)
Total:	**86.2** (85.8-86.6)
FFNN	
Female	**79.5** (78.8-80.1)
Male	**76.2** (75.3-77.0)
Total	**78.3** (77.7-78.9)
XGBoost	
Female	**79.5** (78.4-80.5)
Male	**78.7** (77.7-79.7)
Total	**79.2** (78.3-80.1)

Table 27: Accuracy of algorithms grouped by gender - Learning Fair Representations.

Model	Females (Fall)	Males (Fall)	Females (No Fall)	Males (No Fall)
SVM	31.2	31.8	19.6	20.4
	(30.7-31.7)	(31.2-32.4)	(19.4-19.8)	(20.0-20.7)
LR	52.4	52.9	44.6	44.4
	(51.8-52.9)	(52.3-53.6)	(44.3-44.9)	(43.9-44.8)
RF	47.2	52.6	15.0	15.9
	(46.2-48.1)	(51.4-53.7)	(14.7-15.3)	(15.4-16.4)
FFNN	32.8	33.9	19.3	20.2
	(32.1-33.5)	(33.1-34.8)	(19.0-19.5)	(19.9-20.6)
XGBoost	52.1	56.6	22.6	24.5
	(51.0-53.2)	(55.3-57.8)	(22.1-23.0)	(23.9-25.1)

Table 28: Mean and 95% confidence interval for predicted probabilities conditioned by actual fall and gender. From models trained on LFR data.

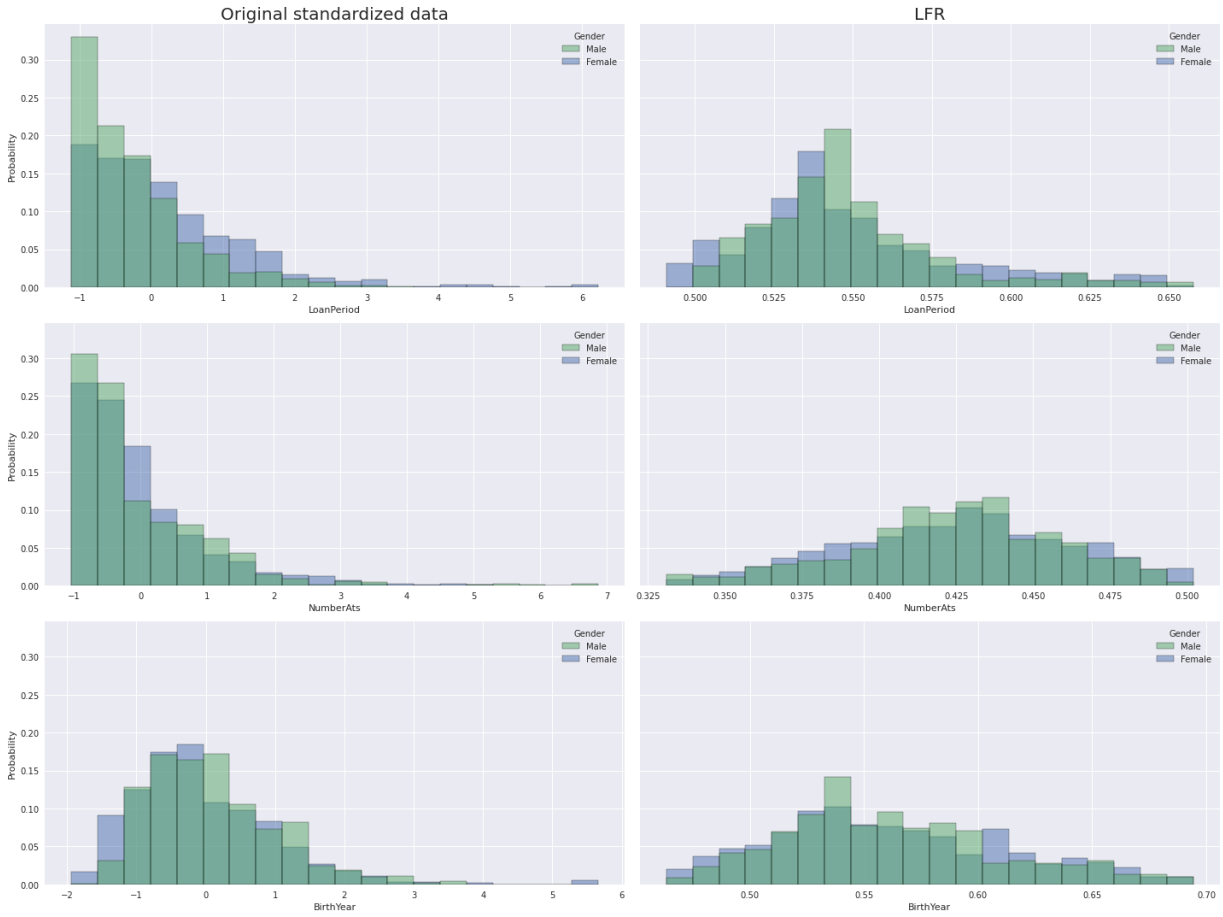


Figure 37: Distribution of numerical features for females and males with LFR.

Model	TP	FP	TN	FN
No mitigation				
SVM	**15.5**	**23.4**	**54.3**	**6.8**
LR	**13.4**	**29.2**	**48.5**	**8.9**
RF	**13.0**	**3.8**	**73.9**	**9.3**
FFNN	**7.9**	**5.7**	**72.0**	**14.3**
XGBoost	**14.2**	**11.8**	**65.9**	**8.1**
Dropping gender				
SVM	**-0.6**	**-1.5**	**+1.5**	**+0.6**
LR	**0**	**+0.3**	**-0.3**	**0**
RF	**-0.3**	**0**	**0**	**+0.3**
FFNN	**-0.3**	**-0.2**	**+0.2**	**+0.4**
XGBoost	**-0.1**	**+1.2**	**-1.2**	**+0.1**
Gender swap				
SVM	**-0.3**	**-3.7**	**+3.7**	**+0.3**
LR	**+0.1**	**+0.1**	**-0.1**	**-0.1**
RF	**+0.1**	**+0.3**	**-0.3**	**-0.1**
FFNN	**+0.3**	**+0.3**	**-0.3**	**-0.2**
XGBoost	**-0.2**	**-0.1**	**+0.1**	**+0.2**
Disparate Impact Removal				
SVM	**-0.7**	**-1.4**	**+1.4**	**+0.7**
LR	**+0.1**	**+0.5**	**-0.5**	**-0.1**
RF	**-1.6**	**0**	**0**	**+1.6**
FFNN	**-0.7**	**0**	**0**	**+0.8**
XGBoost	**-1.5**	**+0.8**	**-0.8**	**+1.5**
Learning Fair Representations				
SVM	**-1.0**	**-0.8**	**+0.8**	**+1.0**
LR	**+0.2**	**+0.5**	**-0.5**	**-0.2**
RF	**-1.1**	**-0.1**	**+0.1**	**+1.0**
FFNN	**-2.7**	**-1.1**	**+1.1**	**+2.8**
XGBoost	**-1.1**	**-0.2**	**+0.2**	**+1.1**

Table 29: Normalised confusion matrix for all models and mitigation methods - with changes in percentages points from the original. Colors indicate: desired (green) and not-desired (red) result.

Model	TP	FP	TN	FN
No mitigation				
SVM	**15.5** (13.6-17.4)	**23.4** (20.9-26.0)	**54.3** (50.5-58.0)	**6.8** (6.1-7.5)
LR	**13.4** (11.5-15.2)	**29.2** (27.2-31.2)	**48.5** (45.3-51.8)	**8.9** (8.3-9.5)
RF	**13.0** (11.7-14.2)	**3.8** (3.6-4.1)	**73.9** (72.4-75.4)	**9.3** (9.1-9.5)
FFNN	**7.9** (7.5-8.4)	**5.7** (5.1-6.4)	**72.0** (71.3-72.6)	**14.3** (13.9-14.8)
XGBoost	**14.2** (13.9-14.5)	**11.8** (11.4-12.2)	**65.9** (65.5-66.3)	**8.1** (7.8-8.4)
Dropping gender				
SVM	**14.9** (13.7-16.0)	**21.9** (21.3-22.5)	**55.8** (54.1-57.4)	**7.4** (7.1-7.8)
LR	**13.4** (12.1-14.7)	**29.5** (29.2-29.8)	**48.2** (46.8-49.5)	**8.9** (8.7-9.1)
RF	**12.7** (11.7-13.7)	**3.8** (3.6-4.0)	**73.9** (72.8-75.1)	**9.6** (9.2-10.0)
FFNN	**7.6** (7.0-8.2)	**5.5** (5.0-6.1)	**72.2** (71.6-72.7)	**14.7** (14.1-15.3)
XGBoost	**14.1** (13.8-14.5)	**13.0** (12.2-13.8)	**64.7** (63.9-65.5)	**8.2** (7.8-8.5)
Gender swap				
SVM	**15.2** (14.0-16.3)	**19.7** (19.1-20.3)	**58.0** (56.2-59.8)	**7.1** (6.7-7.6)
LR	**13.5** (12.2-14.8)	**29.3** (28.9-29.6)	**48.4** (47.1-49.7)	**8.8** (8.7-9.0)
RF	**13.1** (12.1-14.0)	**4.1** (3.8-4.3)	**73.6** (72.5-74.7)	**9.2** (8.8-9.7)
FFNN	**8.2** (7.8-8.7)	**6.0** (5.5-6.5)	**71.7** (71.2-72.2)	**14.1** (13.6-14.5)
XGBoost	**14.0** (13.7-14.4)	**11.7** (10.9-12.6)	**66.0** (65.1-66.8)	**8.3** (7.9-8.6)
Disparate Impact Removal				
SVM	**14.8** (13.6-16.0)	**22.0** (21.4-22.6)	**55.7** (54.1-57.3)	**7.5** (7.2-7.7)
LR	**13.5** (12.2-14.7)	**29.7** (29.1-30.3)	**48.0** (47.0-49.0)	**8.8** (8.7-9.0)
RF	**11.4** (10.2-12.6)	**3.8** (3.5-4.0)	**73.9** (72.8-75.1)	**10.9** (10.6-11.2)
FFNN	**7.2** (6.8-7.7)	**5.7** (5.3-6.2)	**72.0** (71.5-72.4)	**15.1** (14.6-15.5)
XGBoost	**12.7** (12.2-13.3)	**12.6** (11.9-13.4)	**65.1** (64.3-65.8)	**9.6** (9.0-10.1)
Learning Fair Representations				
SVM	**14.5** (13.5-15.5)	**22.6** (21.8-23.3)	**55.1** (53.2-57.1)	**7.8** (7.4-8.2)
LR	**13.6** (12.7-14.5)	**29.7** (29.2-30.2)	**48.0** (47.0-49.0)	**8.7** (8.3-9.1)
RF	**11.9** (10.7-13.2)	**3.7** (3.4-4.0)	**74.0** (72.5-75.6)	**10.3** (10.2-10.5)
FFNN	**5.2** (4.6-5.8)	**4.6** (4.2-5.1)	**73.1** (72.6-73.5)	**17.1** (16.5-17.7)
XGBoost	**13.1** (12.9-13.3)	**11.6** (10.7-12.5)	**66.1** (65.2-67.0)	**9.2** (9.0-9.4)

Table 30: Normalised confusion matrix for all models and mitigation methods. Colors indicate: desired (green) and not-desired (red) result.

B Assessment of classification thresholds

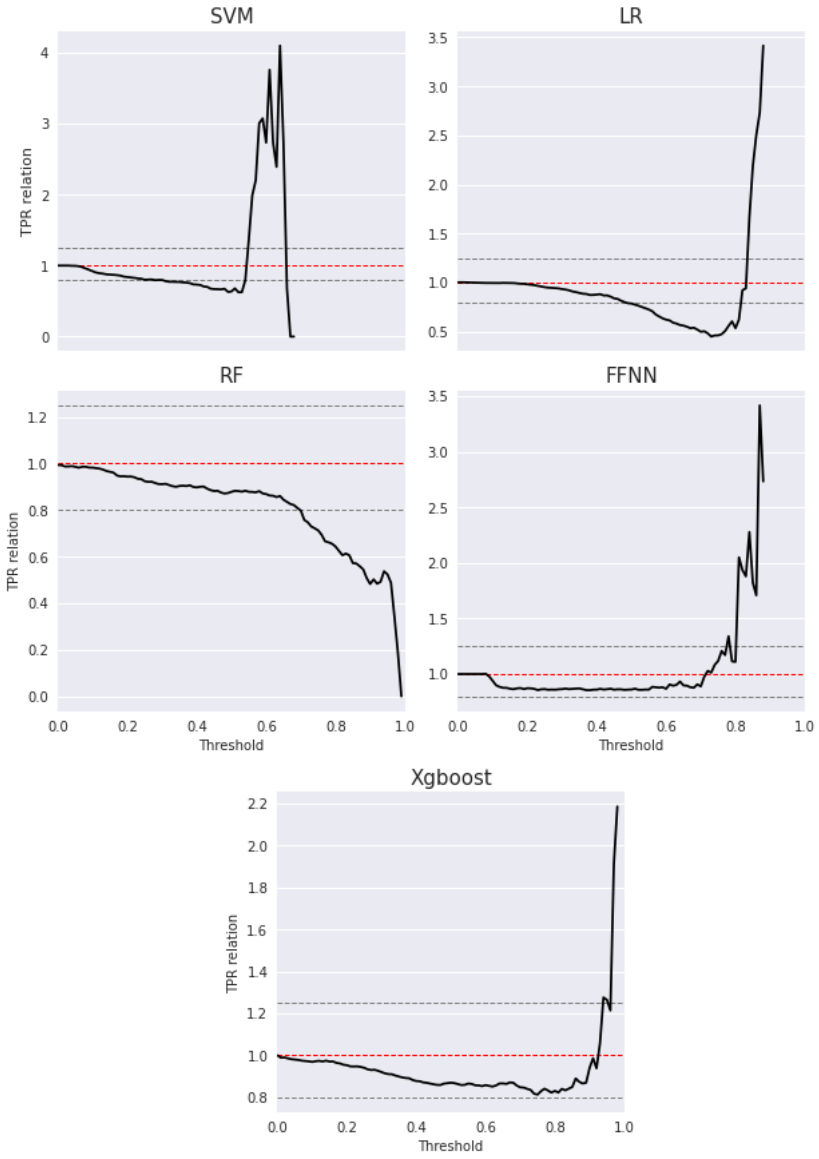


Figure 38: Relation of TPR between female and male vs. threshold for binary classification. Models built on original dataset.

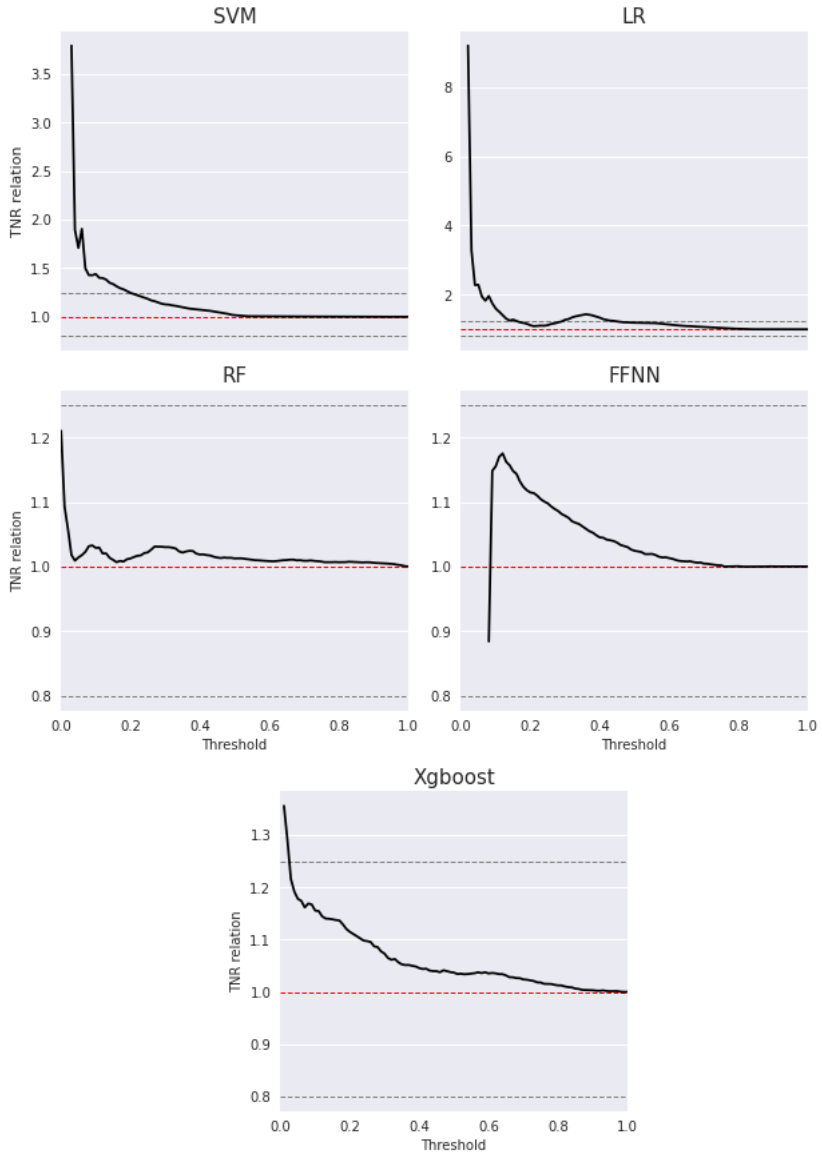


Figure 39: Relation of TNR between female and male vs. threshold for binary classification. Models built on original dataset.

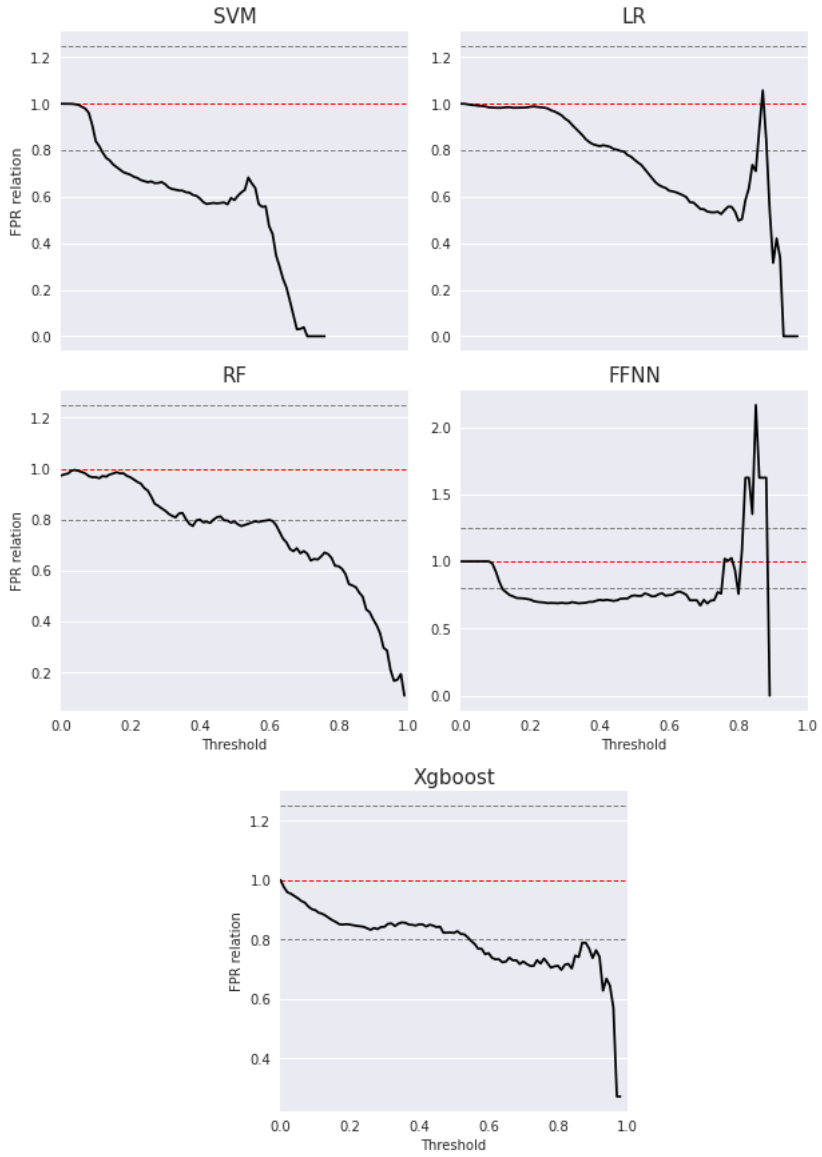


Figure 40: Relation of FPR between female and male vs. threshold for binary classification. Models built on original dataset.

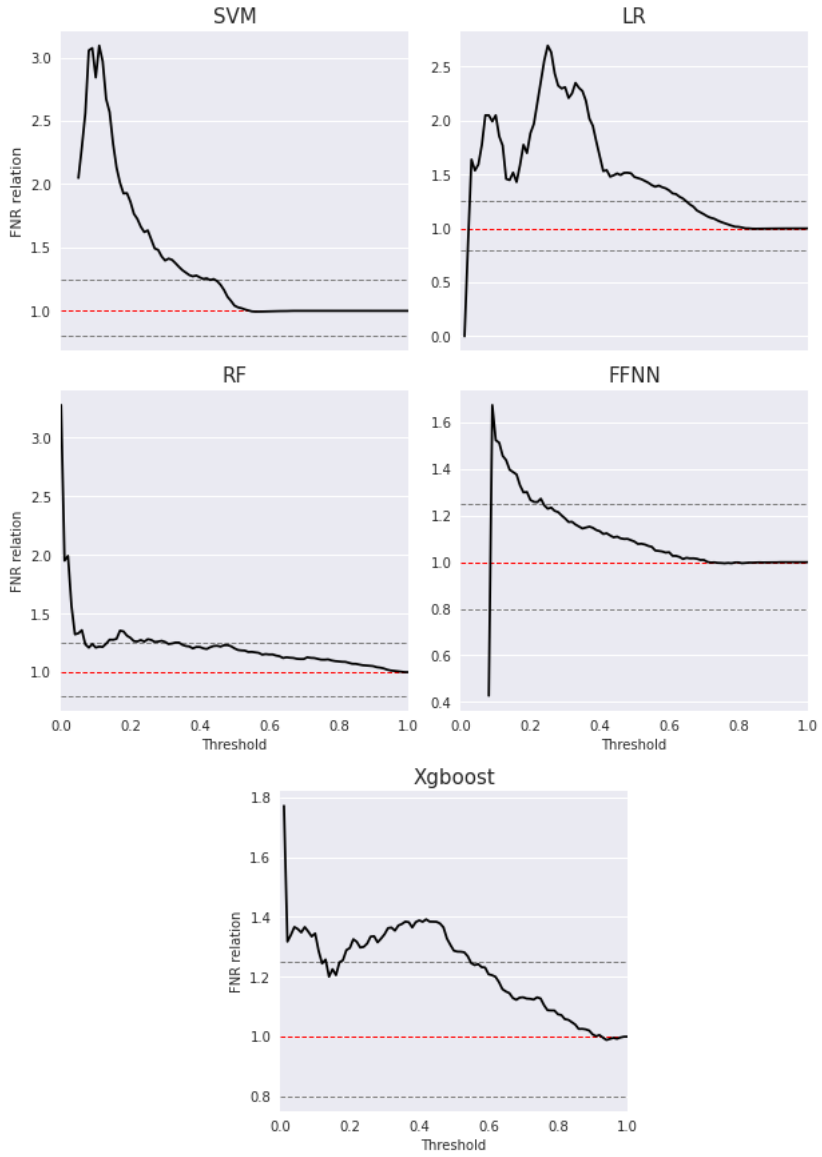


Figure 41: Relation of FNR between female and male vs. threshold for binary classification. Models built on original dataset.

C ROC curves

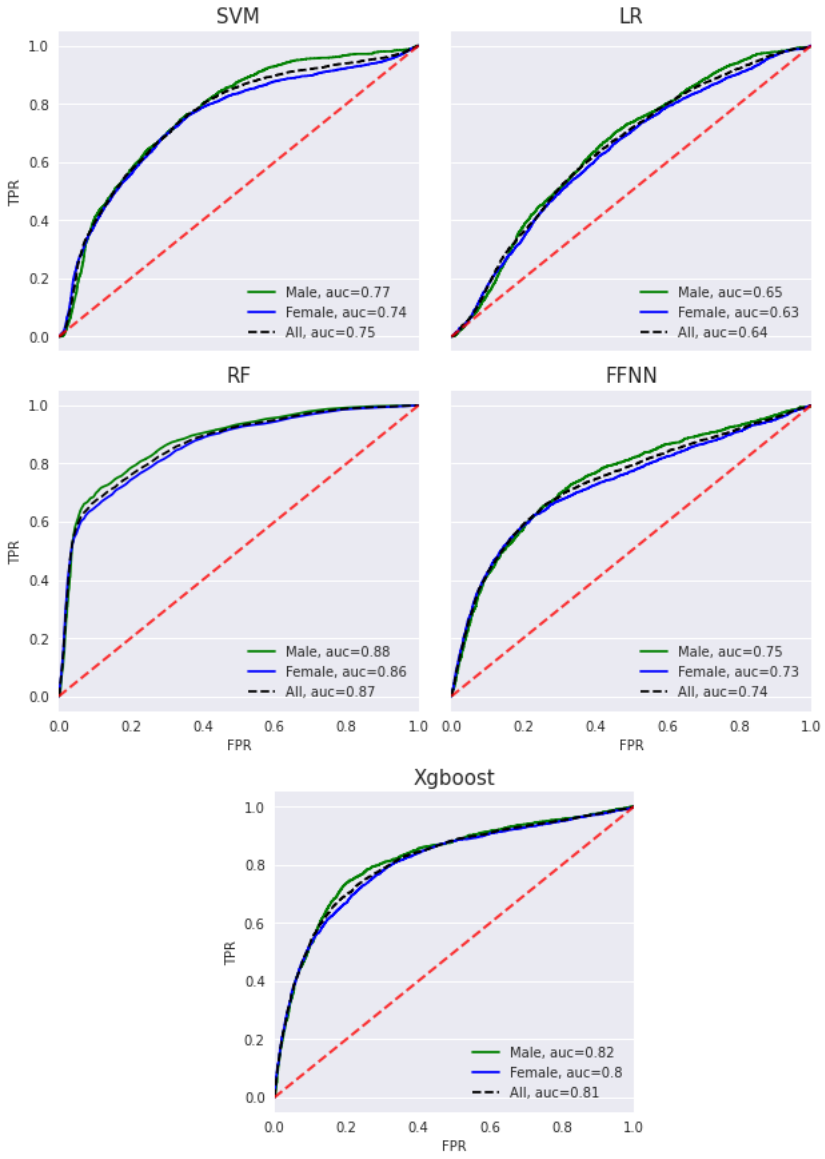


Figure 42: ROC curve and AUC for the models build on the original data.

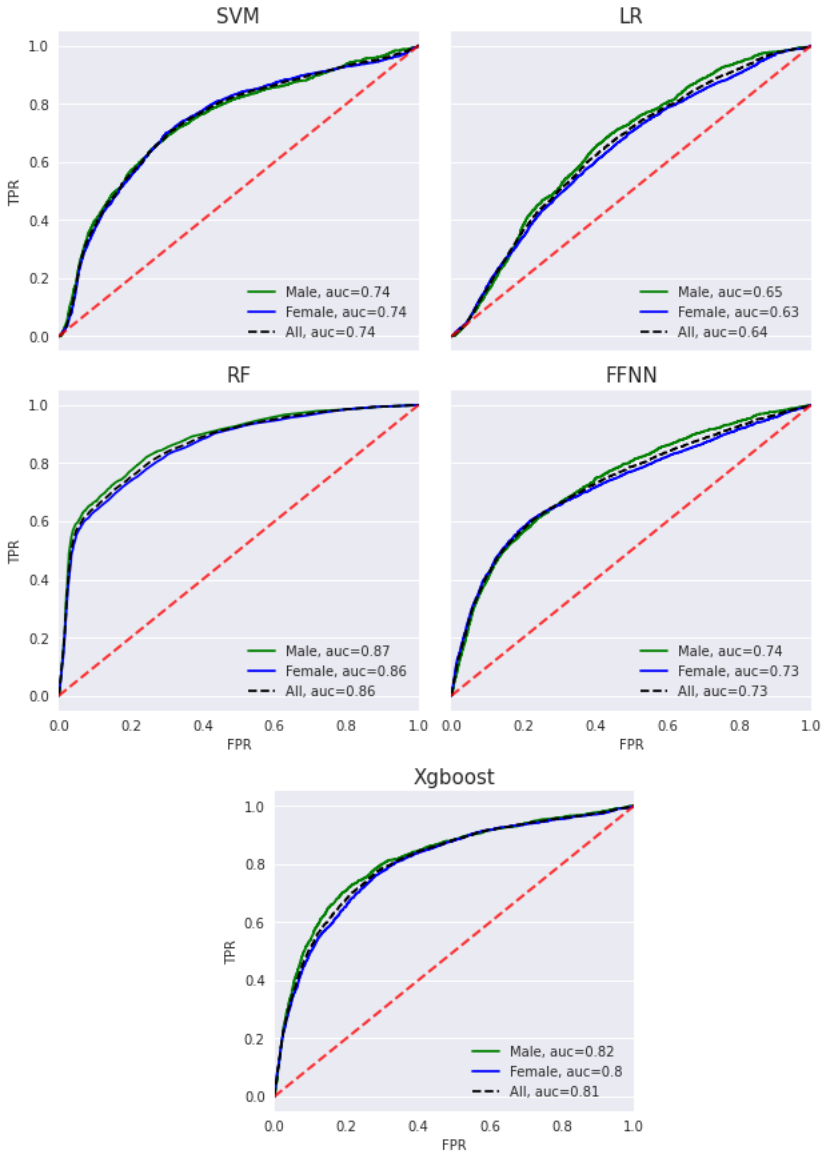


Figure 43: ROC curve and AUC for the models build on the data without gender.

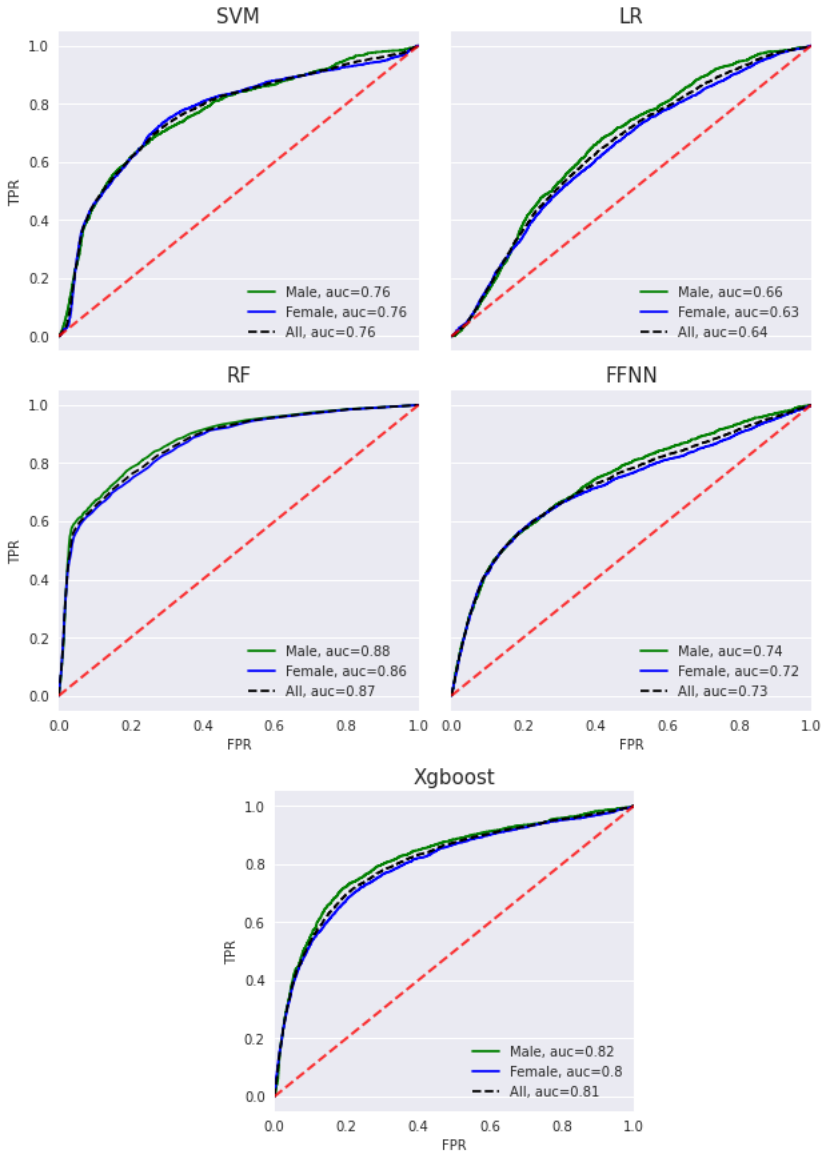


Figure 44: ROC curve and AUC for the models build on the data with gender swap.

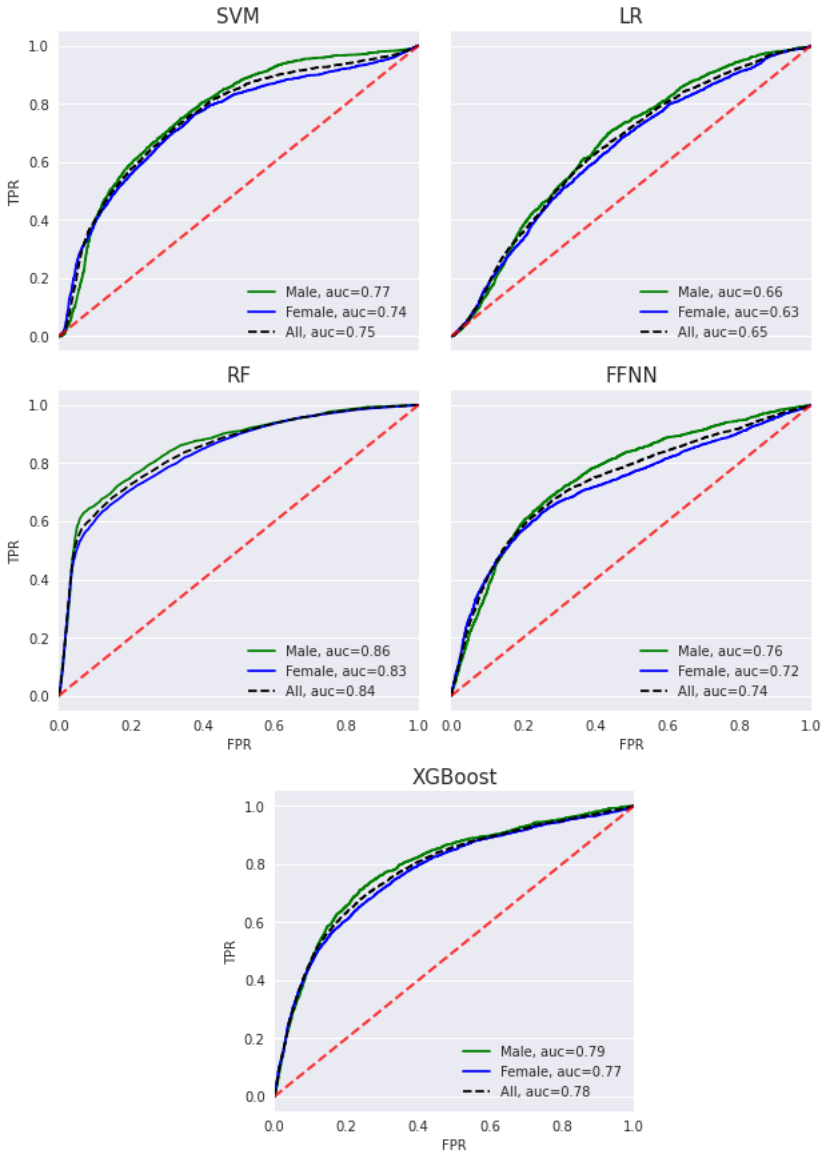


Figure 45: ROC curve and AUC for the models build on the data with disparate impact removal.

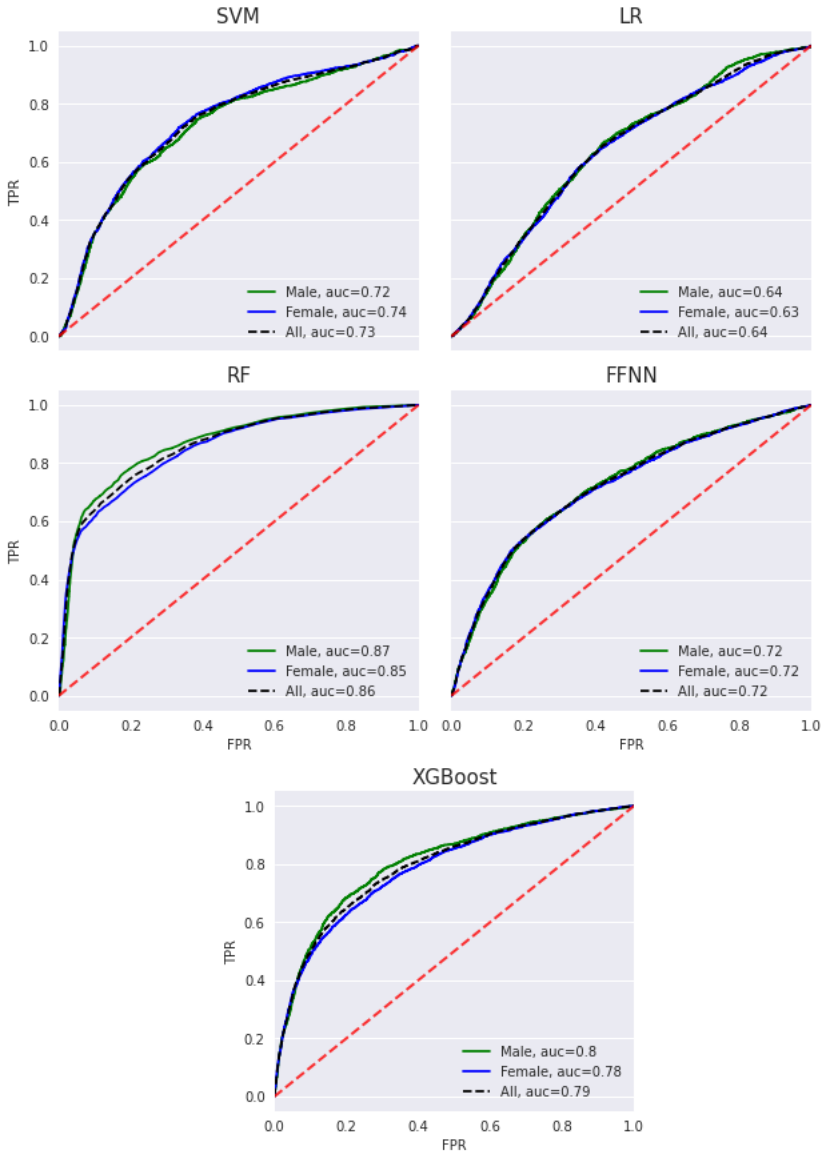


Figure 46: ROC curve and AUC for the models build on the data with LFR.

D SHAP values

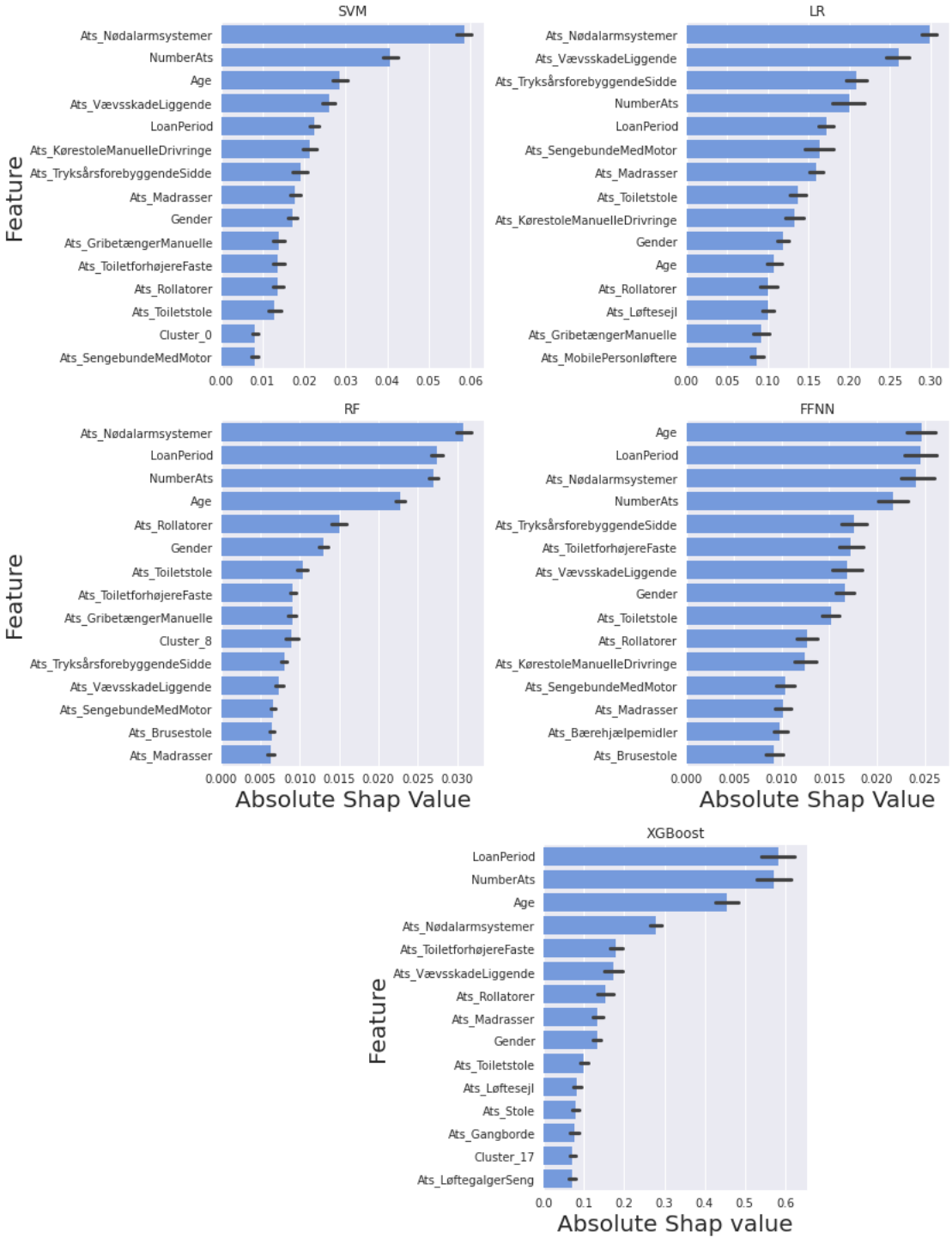


Figure 47: Absolute shap values.

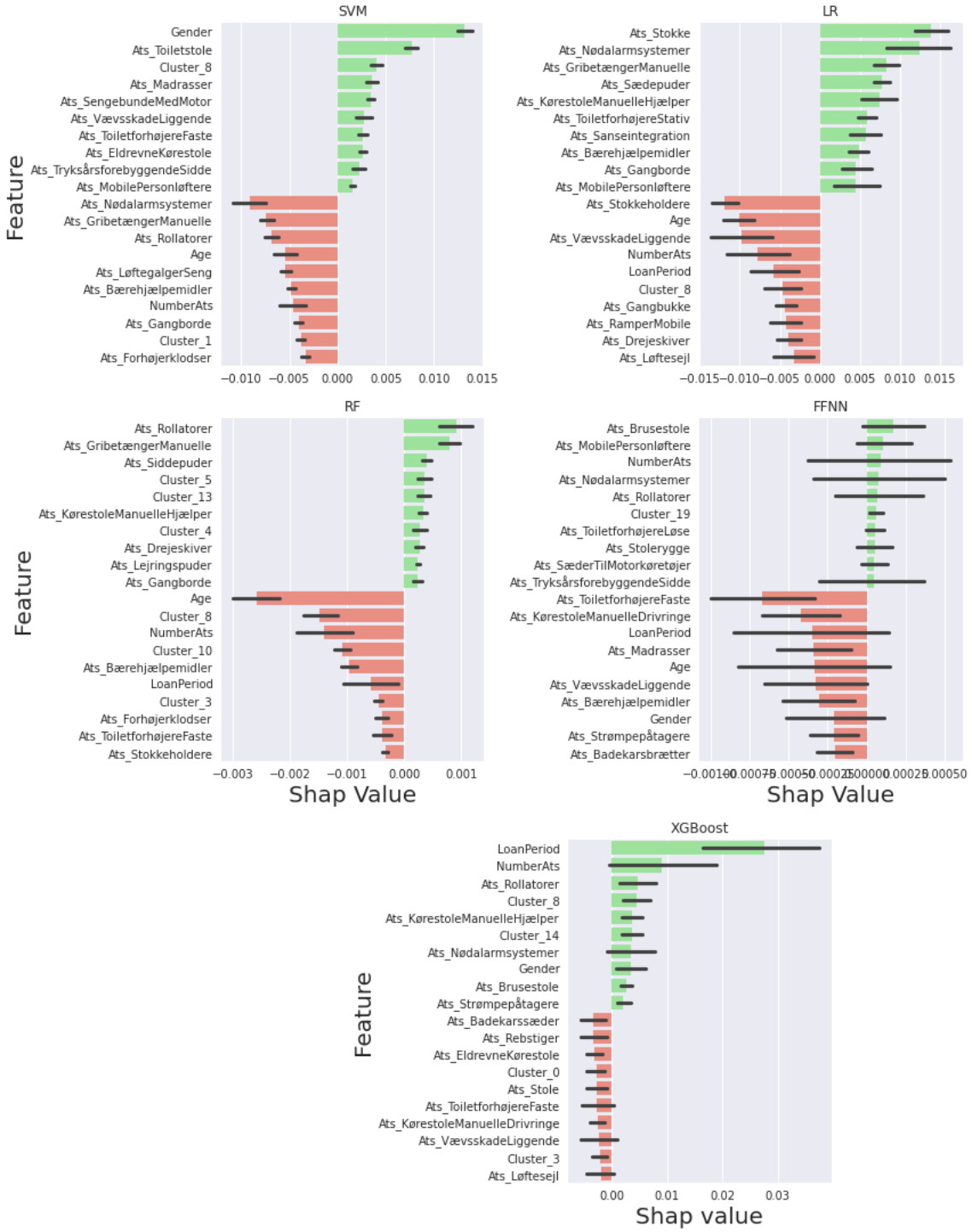


Figure 48: Top 10 negative and positive features based on SHAP value

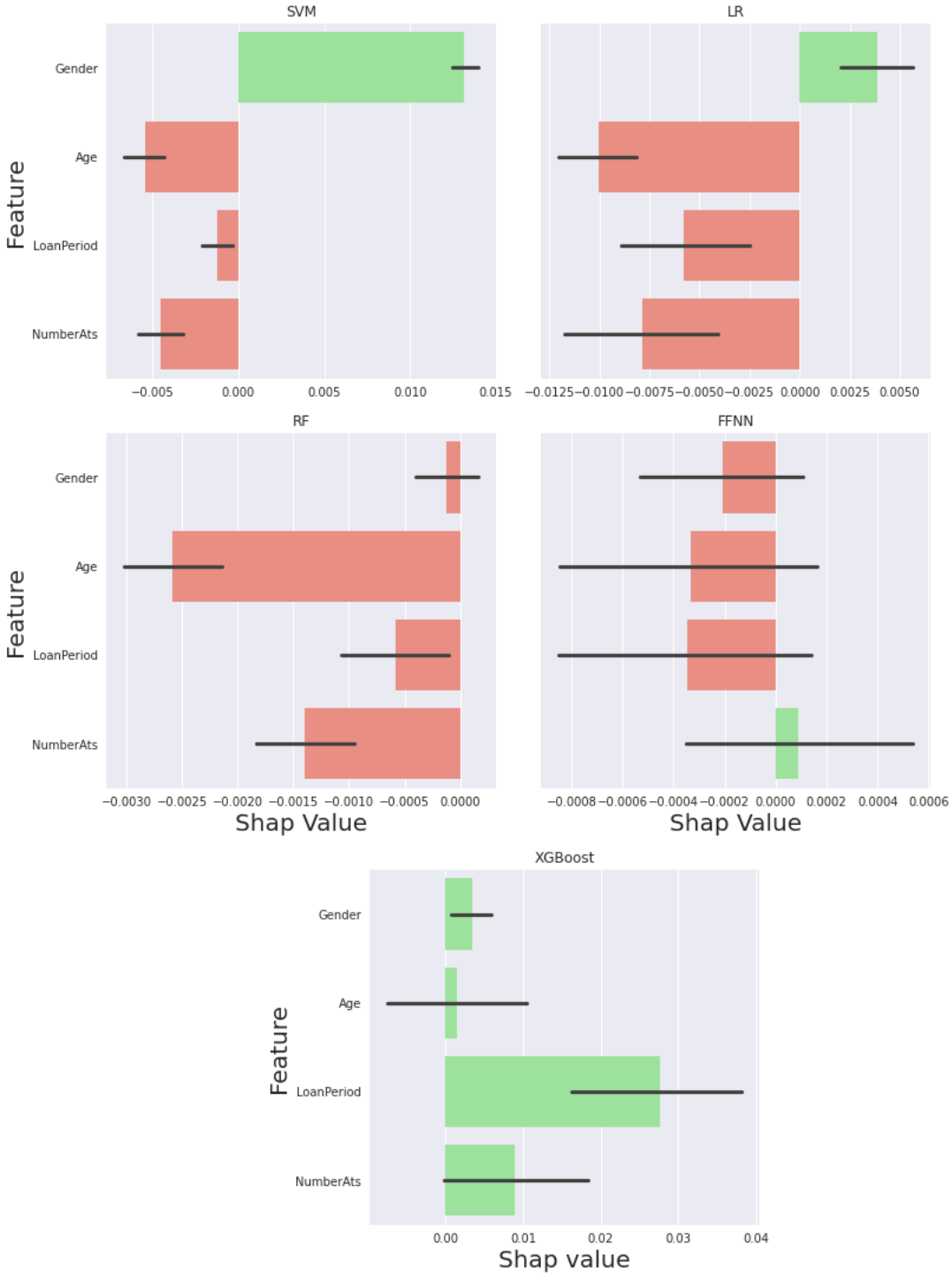


Figure 49: Shap values of the numerical features

E Comparing probabilities across the models

The differences in the distributions of probabilities mean that a somewhat arbitrary choice of model has a substantial impact on the distribution of the predicted probabilities that will be used to predict the risk scores of citizens when the AIR model is implemented. This could easily be overlooked since the intuition behind predicted probabilities is that they are comparable between models - which for these five models on the AIR data set does not seem to be the case. This point can be further explored by showing the range of predictions for a number of randomly drawn citizens from the AIR data set. Table 31 shows the average of the cross validates predicted probabilities and binary classifications for six randomly drawn citizens.

Citizen	Fall	**SVM**	**LR**	**RF**	**FFNN**	**XGBoost**
1	Yes	40.1	57.8	78.2	47.7	87.6
2	No	13.6	38.5	7.9	10.3	29.1
3	Yes	45.7	51.4	78.6	28.0	80.1
4	No	14.3	64.7	12.9	16.4	33.7
5	Yes	47.8	71.2	70.8	39.2	79.4
6	No	9.0	30.2	7.2	11.2	2.1

Table 31: Six citizens' respectively predicted probabilities across all five models.

In table 31 we identify a large span of probabilities across the models for the same citizen. For example, citizen 1, who has fallen, has a range of predicted probabilities from 40.1% to 87.6%. If a caseworker assesses whether citizen 1 should be provided fall training, one could argue that there is a substantial difference between getting a probability score of 40.1% and 87.6%. If the AIR project intends to use the probabilities of the AIR model, it could be relevant to keep this in mind. This is particularly true since it seems that XGBoost has the highest predicted probabilities for those who fall and the second-highest predicted probabilities for those who do not fall, which can be seen in table 10.

F Linear regression analysis

To further nuance our understanding of how the variables are correlated with the probability of falling and how the covariates affect each other, we will build four nested linear regression models, where we iteratively include more features to the covariates used in each model and observe what happens with the coefficients and p-values. We will show the results of all models first and then comment on what we learn about the relations between the variables subsequently. The models are fit to the standardised covariates. We will only comment on the direction of the coefficient and change in p-values of the variables, not on the magnitudes of the coefficient. Furthermore, **we do not** wish to conclude or generalise results from the linear regression analysis, the purpose is simply to obtain a more nuanced understanding of the domain.

We start with the simplest model, that only predicts the fall probability using: *age*, *loan period* and *number of aids*. The coefficients and p-values can be seen in table 32.

Feature	**Coefficient**	**P-value**
Age	0.0211	0.021
Loan Period	-0.0251	0.006
Number of aids	-0.0038	0.683
Adj. R-squared = 0.050		

Table 32: Regression model 1: Only age, loan period and number of aids.

For the second model, the one-hot-encoded aids and clusters are added to the covariates. The result can be assessed in table 33. The coefficients and p-values of the OHE covariate are not shown in the table below for practical reasons (large table) and since the focus of the analysis is the numerical features.

Feature	**Coefficient**	**P-value**
Age	0.0169	0.123
Loan Period	-0.0317	0.008
Number of aids	0.0614	0.079
Adj. R-squared = 0.060		

Table 33: Regression model 2: OHE aids and clusters added.

For the third model, gender is also added to the regression analysis. The result can be seen in table 34.

Feature	**Coefficient**	**P-value**
Gender (1: male)	0.0518	0.016
Age	0.0175	0.111
Loan Period	-0.0276	0.022
Number of aids	0.0618	0.076
Adj. R-squared = 0.063		

Table 34: Regression model 3: Gender added

In the final model, interaction terms between gender and age, loan period and number of aids are added to the model. The results is presented in table 35

Feature	Coefficient	P-value
Gender (1: male)	0.0325	0.159
Age	0.0015	0.907
Loan Period	-0.0160	0.222
Number of aids	0.0732	0.039
Gender (1: male) * Age	0.0493	0.030
Gender (1: male) * Loan Period	-0.0541	0.035
Gender (1: male) * Number of aids	-0.0230	0.385
Adj. R-squared = 0.066		

Table 35: Regression model 4: Interaction terms added

The p-value of **age** increases when adding the OHE aids and clusters in model 2 and remains high when adding gender in model 3 and the interaction terms in model 4. This indicates that the citizen's age does not impact the probability of falling when the aids that the citizen uses are taken into account. However, in model 4, the interaction term between gender and age is positively correlated with falling and has a low p-value. This means that the coefficient describing men's 'return' on the age variable is higher than women's. In other words, compared to women, men seem to be impacted to a higher degree when becoming older in relation to the risk of falling. This is in line with the pattern that could be seen from the box plot in figure 18.

The p-value of **loan period** is low and negatively correlated with the probability of falling for model 1-3, meaning that the longer a citizen has loaned aids on average, the lower the probability of falling. This is contrary to our initial expectation, where we imagined a positive relation between loan period and the probability of falling. On this causal relation, we note that there might be a selection problem, in that those citizens who live long into their senior years are perhaps also more physically and mentally fit. Therefore, at some point, when the less fit elderly - unfortunately - die, those who remain are impacted by their age to a lesser degree. In other words, the mapping between age, loan period and a general notion of physical and mental fitness is not the same across the entire age span. This selection problem might be the cause of the somewhat surprising negative correlation between loan period and the probability of falling. In model 4, where the interaction terms are included, the interaction term between loan period and gender has a low p-value. Here, men have more negatively correlated returns on loan period than women, meaning that loaning aids for a longer period of time diminishes men's risk of falling at a higher rate than women's risk of falling. This result is in line with figure 19, where we saw that those who did not fall had longer loan periods than those who did fall. Again, the interpretation of the causal effect of loan period is not straightforward. Having a long loan period may indicate a long history of physical impairment (high likelihood of falling) or reaching an old age with the help of aids while retaining ones physical abilities (low likelihood of falling). The fact that age and loan period exhibit opposite patterns is surprising - since they both reflect the passing of time in some sense. This could be due to the selection problem mentioned earlier.

The number of aids has a high p-value in model 1 but adding the OHE aid and cluster covariates in model 2 lowers the p-value, and it remains low and positively correlated with the probability of falling in model 3 and model 4. This implies that many aids by themselves are not correlated with falling. However, when considering which specific aids are used by adding the OHE aids, then more aids result in a higher probability of falling. This clarifies the effect of number of aids, which did not have a clear interpretation when assessing the box plots in figure 20. When including the interaction term in the final model, it can be seen that p-value of the interaction term between gender and the number of aids is high. This implies that there are no gender-related differences regarding the returns of number of aids on the probability of falling.

G Qualitative theory of bias in machine learning

This section reviews qualitative approaches for identifying bias and understanding how to asses bias-related challenges in machine learning projects. Since bias has different meanings [61], it can also be assessed in different ways. Furthermore, as the thesis focus on examining machine learning models used for decision-support in the public sector, we find it necessary to remember that *"(...) humans are deeply involved in all parts of the machine learning process"* [15, p. 7].

We expect that the qualitative approaches can create a foundation for a critical reflection of bias evaluation - by not only focus on the technical machine learning aspects - but also assess problems related to the context of the algorithms. The motivation for incorporating qualitative approaches for bias evaluation is greatly expressed by Selbst et. al:

"Achieving fairness in machine learning systems requires embracing a socio-technical view. That is, technical actors must shift from seeking a solution to grappling with different frameworks that provide guidance in identifying, articulating, and responding to fundamental tensions, uncertainties, and conflicts inherent in sociotechnical systems" [59, p. 63].

Selbst et al. emphasize that machine learning systems become a part of socio-technical systems. Achieving fairness in ML requires embracing a socio-technical view on machine learning projects. The thesis evaluates bias in ML algorithms, but since these algorithms potentially become a part of socio-technical systems - grappling frameworks for assessing uncertainties can help support the elucidation of bias in machine learning. Therefore, using qualitative frameworks is anticipated to help to examine both bias in relation to machine learning projects.

Understanding Unintended Consequences of Machine Learning

In their 2020 research paper, Suresh and Guttag identify six different types of bias in the machine learning pipeline [57]. These can be seen in figure 50 below.

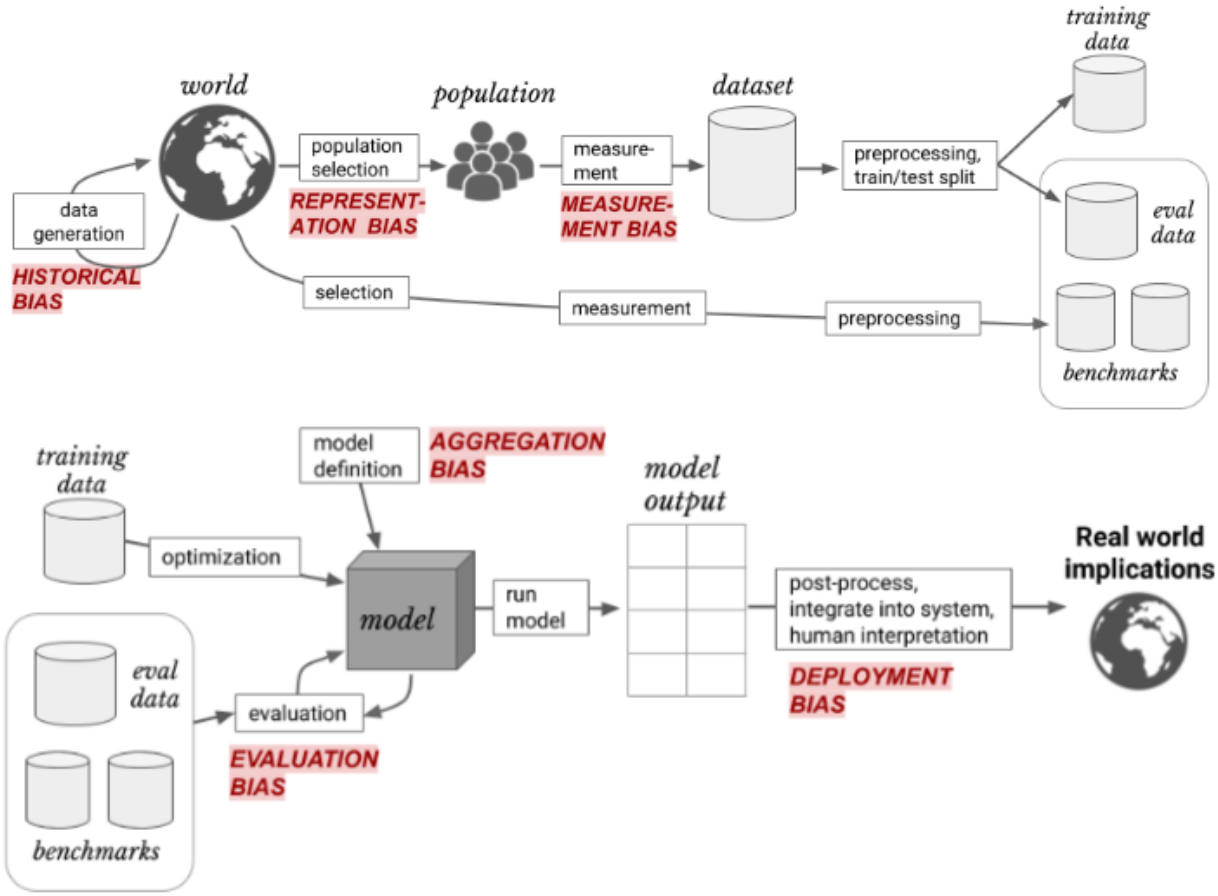


Figure 50: Six types of bias (red) in the machine learning pipeline [57].

Historical bias
Historical bias arises when there is a misalignment between the properties reflected in the world as-it-is and the normative objectives of an ML model, such as historical data reflecting structural racism and an ML-model that tries to avoid reproducing structural racism. This type of bias can exist despite perfect sampling and feature selection. Evaluation of historical bias often involves the examination of representational harm to a particular identity group, such as reinforcing stereotypes [57].

Representation bias
Representation bias relates to the process of defining and sampling the *development population*. The bias arises when the development population under-represents some part of the *use population*, and it can lead to low performance for the group that is under-represented. *Selection bias* is when the probability distribution for the development population is not equal to the distribution of the use population [57]. Representation bias can also occur without *selection bias* when for example, if a minority group makes up only a small proportion of the true distribution. In this case, even sampling in a representative way from true distribution will likely lead to a less robust model for the minority group [57].

As an example, representation bias occurs in the facial recognition case in section 2, where the high misclassification rates of the dark-skinned females could indicate representation bias - here, the data used by the companies had too few dark-skinned females represented.

Measurement bias
Measurement bias has its source in the *choosing*, *collection* and *computing* of features and labels to be used in ML-models. Suresh and Guttag [57] argue that the available variables are typically noisy proxies for the actual features and labels of interest.For example, to measure the label "crime" or a diffuse notion of "riskiness" the proxy variable "arrest" is often used. When choosing, collecting, and computing these proxy variables - which are, in essence, modeling choices - there is a risk of introducing bias into the model. Suresh and Guttag [57] further argue, that measurement bias arises through differences across

groups in *measurement process* and *quality of data*, and finally through *oversimplifications* in defining model tasks.

In the process of *choosing* variables, bias can occur if the choice is inherently associated with noise that is dependent on other variables - for instance, group affiliation.
In the process of *collecting* variables, the *measurement process* can lead to bias.Suresh and Guttag [57] explain how there is a different mapping from crime to arrest in minority communities since they are often more highly policed, which leads to a higher arrest rate. They argue, that this difference in the measurement process led to higher false-positive rates for African American vs. White American defendants in the case of the COMPAS algorithm. Bias can also arise due to differences in *data quality*, where the data quality is dependent on group or individual characteristics.

Finally, measurement bias can arise in the process of *computing*. This could occur through *oversimplifications* during model development, where modeling choices lead to changes in model performances that are dependent on group affiliation, for example, better performance for one group and lower performance for another group.

Evaluation bias
Evaluation bias can arise from using benchmarking/evaluation datasets that are not representative of the intended target population of a model. In the field of data science, models are often compared to one another based on their performance on a commonly known and easily accessible benchmarking dataset, for example, UCI Adult or ImageNet. If these datasets are misrepresentative, benchmarking encourages the development of models that only perform well on a subset of the population. The more popular and widely used a benchmarking dataset becomes, the more severe the issue of evaluation bias will become. To ease the ranking of models, single measure aggregate performance metrics, such as accuracy, are typically used [57].

Aggregation bias
Suresh and Guttag [57] describe aggregation bias as bias occurring when using a one-size-fits-all model on groups with different conditional distributions. In other words, where relationships between features and labels of interest have different group-dependent dynamics. As an example, Suresh and Guttag [57] explains how dynamics regarding the measurement of blood sugar to diagnose and monitor diabetes are known to differ considerably in complex ways across ethnicities and genders. Using a one-size-fits-all approach for modeling can lead to either under-performance for all groups or a model that is fitted to the dominant group. In this sense, aggregation bias is highly intertwined with representation bias. Although, even without representation bias, aggregation bias can still lead to harmful outcomes.

Deployment bias
Deployment bias relates to a mismatch between what the model is intended to solve and how the model is actually used. The bias often occurs when a model built for fully autonomous tasks becomes part of a socio-technical system influenced by human decision-makers. This influence can be seen when inappropriate interpretation or usage of model predictions arise, for example, when human decision-makers use algorithms for *off-label* purpose - i.e., another purpose than originally intended. [57].

Relevance to the thesis
The six types of bias identified in the machine learning pipeline could be a useful framework to structure critical reflection regarding the entire process of the AIR project. Suresh and Guttag show how bias can arise from many different sources and in many different ways and that not all of these are quantifiable. This is the case for historical bias, where the data itself reflects an unwanted bias. Addressing the bias types listed in the paper [57] could be fruitful to think about bias in a more holistic way than it is possible with just statistical approaches.

Traps in fair-ML

Selbst et al. [59] outline five "traps" regarding fairness that one can fall into when working with machine learning projects. The paper argues that fairness is a property of social and legal systems, like employment and criminal justice, and fairness is not a property of the technical tools. To treat fairness the paper identifies the following five traps:

- Framing trap
- Portability trap
- Formalism trap
- Ripple effect trap
- Solutionism trap

These are traps that result from failing to properly account for or understand the interactions between technical systems and the social world. In the following, there is a brief overview of the traps.

The framing trap
The most common abstraction of machine learning consists of choosing representations (data) and labeling (of outcomes). The choices made constitute the *algorithmic frame*. In this frame, the efficacy of a model is evaluated based on the relation between the input and the output. The authors state that in the algorithmic frame, notions of fairness cannot be defined. Therefore, to investigate fair machine learning, it requires one to also encompass the *data frame* - which are the input and output of an algorithm - and how do these affect the model. Finally, the *socio-technical frame* recognizes that the model is a part of a socio-technical system. Selbst. et al. distinguish between the output of the ML model and the output of the socio-technical system the ML model is a part of. It is therefore important also to assess the human outcomes (decisions) for determining fairness [59].

The framing trap illustrates how projects can fail to model the entire system that covers all relevant aspects of a particular phenomenon of interest. More specifically, when modeling any given subject, researchers will use a frame to capture and measure relevant aspects and define the scope of the problem. When defining the scope and limits of the model, some relevant aspects could be left out. Depending on the specific aspects left out, failure to model the entire system might lead to bias or inadequate evaluation of fairness.

The portability trap
This trap emphasizes how using an algorithm designed for one social context maybe be misleading or inaccurate when applied in another context. A reason to fall into this trap is that computer science culture prizes and often demands portability. Transferable code, purposefully designed to be as abstract as possible, is considered more useful (because it is reusable). Portability allows the same "solution" (for example, a better algorithm for binary classification) to apply to problems in various social settings. Certain assumptions will hold in some social contexts but not others. The assumptions should reflect the anticipated application. [59]

The formalism trap
The formalism trap emphasizes that machine learning projects can fail to account for the whole meaning of fairness - therefore, this concept cannot be solved through mathematical formalisms. Since algorithms "speak math", there has been a tendency to mathematically define aspects or notions of fairness in society to incorporate fairness ideals into machine learning. Definitions in the literature of mathematical fairness are simplifications that cannot capture the full range of similar and overlapping notions of fairness and discrimination in philosophical, legal, and sociological contexts. The paper refers to legal scholar and philosopher Deborah Hellman and states that *"we make distinctions all the time, but only cultural context can determine when the basis for discrimination is morally wrong"* [59].

The ripple effect trap
If machine learning projects do not understand how the new technology changes the behavior of a pre-existing system, the projects end up in the ripple effect trap. Implementation of technology in a social context both has intended and unintended consequences. Unintended consequences could be how people or organizations in a system will respond to the new technology. For example, when using a risk assessment machine learning tool that produces a score for the probability of a criminal to do more crime: How does it affect the judge? Will the judge suddenly only use the score and neglect other factors? When seeing the scores frequently, could that change how the scores are interpreted through time? [59]

The solutionism trap
In the best case, machine learning developers can improve a model, which encompasses more and more social context, so the model can approximate the social environment. However, as [59] states, by starting from technology and working outwards, there is never an opportunity to evaluate whether the technology should be built in the first place [59]. The solutionism trap happens when researchers or practitioners fail to recognize that a solution to solve a problem may not involve technology.

Selbst et al. [59] state two broads situations where the solutionism trap appears: 1) when fairness definitions are politically contested and 2) when modeling is too complex to be computationally tractable. In the area of predicting politics, the authors emphasize how difficult a task it is since human preferences are not rational and human psychology is not conclusively measurable.

Addressing the traps
The main takeaway from the paper is the process of determining and applying technical solutions. The main take-aways will here be represented as five questions that could be asked when building fair ML solutions [59]:

- Solutionism trap: Is the technical solution appropriate in the first place?
- Ripple effect trap: Does the technical solution affect the social context in a predictable way such that the social context, that is intended to stay unchanged, remains unchanged after the introduction of the solution?
- Formalism trap: Can the technical solution appropriately handle robust understandings of social requirements such as fairness?
- Portability trap: Has the technical solution appropriately modeled the social and technical requirements of the actual context in which it will be deployed?
- Framing trap: Is the technical solution heterogeneously framed so as to include the data and social actors relevant to the localized questions of fairness?

Relevance to the thesis
The traps provide a list of typical mistakes researchers and practitioners make when implementing machine learning models or evaluating bias and fairness in relation to these models. One could easily imagine that the AIR project is challenged in similar ways, and the traps described in [59] therefore enables our evaluation of the AIR project to focus on aspects of the implementation that are more likely to be problematic in relation to bias and fairness.

Algorithm hygiene

Lee et al. [62] suggest a framework for *algorithm hygiene* - which identifies some specific causes of biases and employs best practices to identify and mitigate them. The paper draws upon the insight of 40 thought leaders from across academic disciplines, industry sectors, and civil society organizations who participated in one of two roundtables.

The paper focuses on two causes of bias: historical human bias and incomplete/unrepresentative data. Overall, the paper emphasizes that pervasive and often deeply embedded prejudices shape historical human biases against certain groups, which can lead to their reproduction in computer models. Incomplete/unrepresentative data causes algorithmic bias, for example, if the training data over-represents a group, the model may systematically predict worse for the under-represented group(s). [62]
For mitigating bias, the paper proposes that operators of algorithms must develop a bias impact statement. The impact statement can help probe and avert any potential biases that are baked into or are resultant from the algorithmic decision. In order to develop the statement, the paper offers a template of questions. These questions can guide, for example, developers through the design, implementation, and monitoring phases.

They propose that operators apply the bias impact statement to assess the algorithm's purpose, process, and production, where appropriate. The proposed bias impact statement starts with a framework that identifies which automated decisions should be subjected to, such as scrutiny, operator incentives, and

stakeholder engagement. Then, the user incentives should be addressed. Finally, the stakeholders should be engaged.
This forms three elements or questions:

- Which automated decisions?
- What are the user incentives?
- How are stakeholders engaged?

The elements are reflected in a set of in-depth questions that operators should answer during the design phase to filter out potential biases - it is called the bias impact statement (see table 36).

In a final remark about the paper, the roundtable participants emphasized the importance of cross-functional and interdisciplinary teams to create and implement the bias impact statement. Operators of algorithms should seek to have a diverse workforce team. Employing diversity in the design of algorithms upfront will trigger and potentially avoid harmful discriminatory effects on certain protected groups. [62]

What will the automated decision do?
Who is the audience for the algorithm and who will be most affected by it? Do we have training data to make the correct predictions about the decision? Is the training data sufficiently diverse and reliable? What is the data lifecycle of the algorithm? Which groups are we worried about when it comes to training data errors, disparate treatment, and impact?
How will potential bias be detected?
How and when will the algorithm be tested? Who will be the targets for testing? What will be the threshold for measuring and correcting for bias in the algorithm, especially as it relates to protected groups?
What are the operator incentives?
What will we gain in the development of the algorithm? What are the potential bad outcomes and how will we know? How open (e.g., in code or intent) will we make the design process of the algorithm to internal partners, clients, and customers? What intervention will be taken if we predict that there might be bad outcomes associated with the development or deployment of the algorithm?
How are other stakeholders being engaged?
What's the feedback loop for the algorithm for developers, internal partners and customers? Is there a role for civil society organizations in the design of the algorithm?
Has diversity been considered in the design and execution?
Will the algorithm have implications for cultural groups and play out differently in cultural contexts Is the design team representative enough to capture these nuances and predict the application of the algorithm within different cultural contexts? If not, what steps are being taken to make these scenarios more salient and understandable to designers? Given the algorithm's purpose, is the training data sufficiently diverse? Are there statutory guardrails that companies should be reviewing to ensure that the algorithm is both legal and ethical?

Table 36: Design questions template for bias impact statement from [62].

Relevance to the thesis
In terms of sources of bias, the paper by Lee et al. [62] resembles the notions of historical bias and representation bias from the Suresh & Guttag paper [57]. Lee et al., however, also propose a bias impact statement that specifically targets sources of bias that practitioners can address and take action on before creating algorithms to make automated decisions, or in the AIR case, be used for decision support. In the AIR case, the questions in the bias impact statement can be used as a guide for questions that lead to actionable insights, for example, if an analysis of the incentives of users, for example, the caseworkers, presents reasons to change certain aspects of the algorithm in relation to bias.

The trouble with bias

In 2017 Kate Crawford spoke at the NeurIPS conference about "Trouble in bias" [61]. Crawford distinguishes between two types of harms related to bias: *harms of allocation* and *harms of representation*. These harms have the following characteristics:

Allocation	Representation
Immediate	Long term
Easily quantifiable	Difficult to formalize
Discrete	Diffuse
Transactional	Cultural

Table 37: Characteristics of allocation and representation [61]

Allocation can be characterised as being immediate - since it is a time-bound moment of decision making. Furthermore, it is often quantifiable. This raises questions about fairness and justice in discrete and specific transactions. Representation is a long-term process that affects attitudes and beliefs. Thus, it can be more difficult to formalize. Representation is about a more diffuse depiction of humans and society - thus, representation is cultural.

Harms of allocation
Harms of allocation is when a system allocates or withholds certain groups an opportunity or resource. These types of harms are often economically oriented and centered on harms related to decisions regarding, for example, who gets loans or gets insurance.

Harms of representation
Harms of representation refer to machine learning as being harmful regarding the representation of human identity - not only a system for contributing to decision making. The harms occur when systems reinforce the subordination of some groups along the line of identities such as race and gender. This sort of harm can occur regardless of whether resources are being withheld from members of a protected class.

Relevance to the thesis
To better understand the harms of potential bias in the AIR project, Crawfords distinctions can help assess whether an ML algorithm is a part of a decision-making process where the public authority, for example, has to allocate resources. If so, the characteristics of allocation harm can help uncover, for example, the time perspective of the harms. Also, by understanding how allocation harm are related to specific transactions, it will help question how fairness relates to the transactions - which in the AIR case are whether a citizen should be provided fall-training. The characteristics of harms of representation could be beneficial in a critical reflection of how the AIR machine learning model and related data can, in the long term, for example, reinforce stereotypes.

Crawford's [61] two types of harms can be associated with some of the elements of Suresh and Guttag's framework [57]. For example, the harms of allocation can be related to evaluation bias since the fairness of resource allocation can be viewed differently depending on the evaluation measure. Representational harm can, for example, be related to representation bias, since under-representation in the development population can lead to low performance of a group of subjects - for example, if facial recognition underperforms and do denigration harm towards the under-represented group (as seen with the darker females in section 2). Harms related to algorithms, like the two types Crawford addresses, are also examined by Lee et al. [62] since the bias impact statement also helps outline, for example, who can be harmed by the system and which potential bad outcomes could occur.

Data Statements for Natural Language Processing

Bender & Friedman [14] propose data statements as a way of bringing about change in the field of natural language processing (NLP) towards more ethically responsible use of NLP technology. Bender & Friedman have designed the data statement framework explicitly as a tool for mitigating bias in systems that use language data for training and testing.

Bender & Friedmans proposed data statement framework consists of two parts: a long-form schema and a short form. The long-form statement consists of several categories with relevant information that should be created for any new dataset containing natural language data. In contrast, the short form statement should be included in any publication using a dataset. The short-form statement is envisioned as a 60-100 word summary of the long-form statement. The long-form statement consists of the following elements:

- Curation rationale: Which texts were included, and what was the goal of obtaining them?
- Language variety: Standardised language tags, along with prose descriptions of relevant varieties, for example, English as spoken in Palo Alto, California.
- Speaker demographic: Speaker is defined as the source of the natural language data and should include information about their age, gender, race/ethnicity, native language, socioeconomic status, number of speakers, and presence of disordered speech.
- Annotator demographic: Annotators are the people who have marked the data with relevant tags/labels, and information about their age, gender, race/ethnicity, native language, socioeconomic status, and training in linguistics should be included.
- Speech situation: This includes time/space, modality (spoken, written), edited or spontaneous, synchronous or asynchronous interaction, intended audience.
- Text characteristics: Genre and topics.
- Recording quality: If audiovisual recordings are present, the quality can be commented.

These elements are essential when considering whether the data used is appropriate for NLP work being undertaken and to ensure that the reported results are properly contextualised [14].

Relevance to the thesis
In the context of this master thesis, that does not have en exclusive focus on NLP, the underlying notions behind the data statement framework are still relevant. Here, a data statement that describes general characteristics in categories inspired by the specific NLP-related categories of the Bender & Friedman paper could be a way to explicitly describe data set characteristics that could lead to bias when using algorithms trained or tested on specific data sets. Since the paper presents a set of elements that practitioners should consider when thinking about bias, there are similarities with the bias impact statement shown in [62] section G, although the Bender & Friedman data statement focuses more on the characteristics on the data and the curation process of the data set, rather than the general context of the algorithm.

Bias in Machine Learning - What is it Good for?

In [15] Hellström et al. survey the current literature on bias in machine learning, and analyse and discuss how different types of bias are connected and depend on each other. Figure 51 is an illustration of the different categories in the taxonomy and where they are placed in the machine learning pipeline.

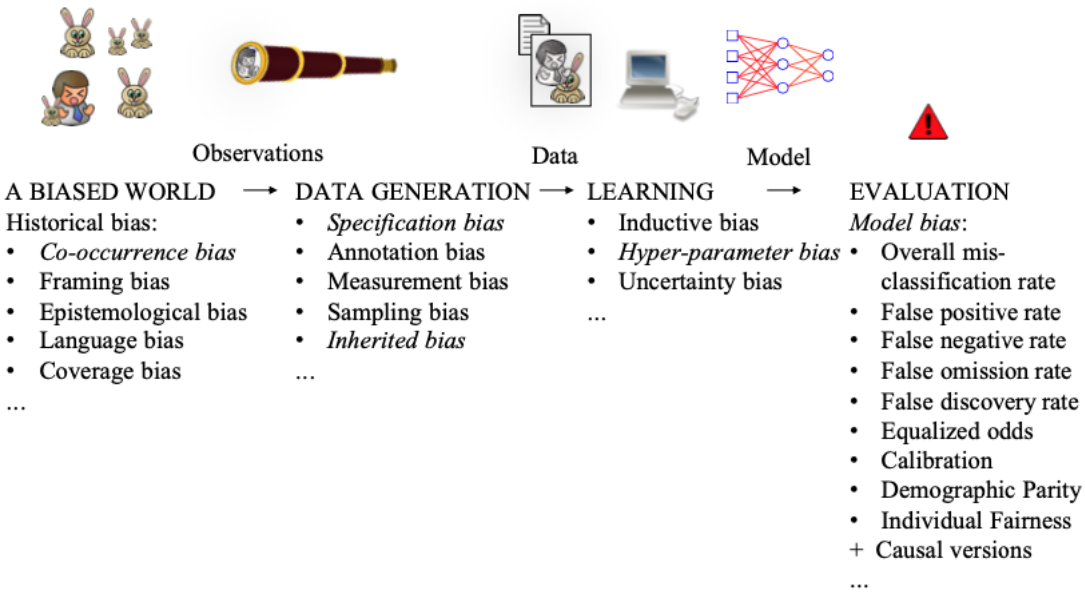


Figure 51: Taxonomy of bias in machine learning in [15]

Hellström et al. underline a distinction between *model bias* and the rest of the bias types from figure 51. While all other bias types describe the sources of potential bias in machine learning systems, model bias refers to the detected bias in the outcome of these systems and how they are defined (for example, false-positive rates or group fairness). The four categories: *biased world*, *data generation*, *learning* and *evaluation*, are briefly covered in the following:

- A biased world: Refers to sources of bias that come from the world *as it is/was*, meaning that these biases would be propagated into the model outcome even without biased procedures from data generation or learning. This *historical bias* comes from human actions and historical outcomes that manifest themselves in observations that have unwanted properties and can be expected to be learned by any machine learning model since most machine learning techniques mimic the observations of the world.
- Data generation: Refers to acquiring and processing observation and turning it into data for a machine learning model. Here bias can occur from, for example, choosing input and output variables of the learning task, measurement errors, or sampling.
- Learning: Bias in the learning step comes from choices regarding how the model learns and outputs, for example through testing a limited set of hyper-parameters or by setting thresholds for acceptable levels of uncertainty.
- Evaluation: Refers to model bias detected in the outcome of the machine learning system. However, this bias can be defined in many different ways, many of which are related and some of which are conflicting and cannot be avoided simultaneously. In most assessments of model bias, the bias metrics are used to assess differences in classification between protected attributes, such as gender, race, ethnicity, religion, etc.

Hellström et al. [15] note that to be able to assess the true extent of the bias for any machine learning system, one must define a notion of how the world *should be*, as opposed to how the world is or was. This notion is, as an example, built into the group fairness view on the bias, where the model output should be independent of a protected attribute. In other views, for example, individual fairness, where the model's outputs for two similar individuals should be the same, there is a different notion of how the world should be. These difference are at the heart of many discussion regarding bias and fairness, for example regarding the COMPAS algorithm as presented earlier in 2.

Relevance to the thesis

Overall, Hellström makes a distinction between the term "model bias", which relates to the outcome of

the system, and all other types of bias. The division of the machine learning pipeline can, to some extent, be related to Suresh and Guttag [57] framework. "A biased world" relates to "historical bias," which together represent how framing and history influence (bias) the input data. "Data generation" relates to "representation bias" / "measurement bias" and focuses on the data sampling and collection process. The "learning" category is related to "aggregation bias" - both categories assess how the model is defined or optimized. Finally, Hellström's "evaluation" category relates to Suresh and Guttag's "evaluation bias", which both assess that bias occurs in evaluating the model performance and the associated metrics.

Furthermore, Hellström's "learning" category can help put focus on the role of the ML developer. When building models that reproduce the COMPAS algorithm bias, different hyperparameters have been chosen. Investigating or at least discussing what influences the learning category's bias could have on an ML project (or, for example, on the model bias) could be interesting to examine. Also, Hellström emphasizes that in order to assess bias, one must define how the world should be.

H The COMPAS algorithm

Method considerations

The COMPAS dataset is included in the thesis since the COMPAS case is often used in the machine learning literature as an example of an algorithm where the classifications exhibit racial bias. Since there is a well-documented expectation of finding bias in the COMPAS case [7], it is a meaningful data set to develop and implement methods for bias identification and mitigation. Furthermore, the COMPAS case is similar to the AIR case in the following important ways.

The COMPAS algorithm is used to predict whether a defendant re-offends or not - therefore, it is a binary classification problem. This corresponds to the AIR case which is also a binary classification problem, predicting if a citizens will fall or not. Furthermore, the COMPAS algorithm is used by judges in the US, while the AIR algorithm is anticipated to be used by caseworkers in Aalborg Municipality [2], making them both decision support tools used in the public sector. Finally, the two cases are similar in the sense that they are used to assess if a citizen should be allocated or withheld an opportunity or resource, respectively pretrial release and fall-prevention training.

Reproducing the COMPAS bias

The COMPAS algorithm is closed-sourced [7], and therefore, it is not possible to test bias identification and mitigation techniques on the COMPAS algorithm itself. Because of this, we implement machine learning models that attempt to reproduce the COMPAS algorithm's bias, on which we can test the bias identification and mitigation techniques. If the techniques mitigate bias in our implementations, it cannot be guaranteed that it will also mitigate bias for the COMPAS algorithm, but it can nonetheless be used as an experimental result that strengthens the relevance and application of the techniques.
We have chosen to built the same four machine learning models as we intend to built on the AIR data set. See section 8.3 for details of why we choose the models.

Background

The COMPAS algorithm is a commercial software tool for decision support used in the United States justice system, for example, in the State of New York and the State of California, among others [63]. The software assesses a criminal defendant's likelihood of becoming a recidivist - which is a term that describes whether a criminal re-offends [36]. Judges use the COMPAS score as a decision support tool when determining pretrial release, and sentencing [6]. In 2016 ProPublica published the article "Machine Bias", where the authors investigated racial disparities in predictions of the COMPAS software [7].
ProPublica analysed data from more than 7000 criminal defendants in Broward County, Florida and compared the predicted recidivism scores with the actual recidivism rates over a two-year period, the main findings were the following:

- African-American defendants were often predicted to be at a higher risk of recidivism than they actually were (nearly twice as likely compared to Caucasian defendants)
- Caucasian defendants were often predicted to be less risky than they were (nearly twice as often as African-American defendants)

The analysis showed that the algorithm correctly predicted an offender's recidivism 61 % of the time [7].

COMPAS data overview

The COMPAS dataset has 7214 records of defendants' COMPAS scores and criminal history. The dataset both contains information about defendants' criminal history before and after they got a COMPAS score. The original dataset has 52 columns, including information about the charge that resulted in the COMPASS scoring and the charge committed after (if any). The charges (before and after) had associated data describing the date of the crime, the date when the defendant was jailed and got out.

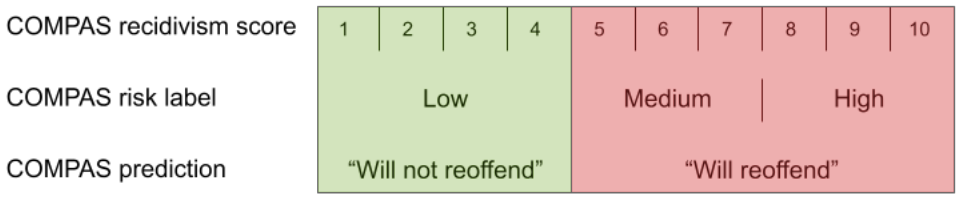


Figure 52: COMPAS scores and risk labels. Illustration by Daniel Juhász Vigild and Lau Johansson

Each pretrial defendant received a COMPAS scores: “Risk of Recidivism”. As illustrated in Figure 52 COMPAS scores for each defendant ranged from 1 to 10, with ten being the highest risk. Scores 1 to 4 were labeled by COMPAS as “Low”; 5 to 7 were labeled “Medium”; and 8 to 10 were labeled “High” [36]. As shown in the illustration, "Low" lead to the prediction "Will not reoffend" while "Medium" and "High" leads to the prediction "Will reoffend".

The confusion matrix tables below show the classifications made by the COMPAS algorithm and the recidivism ground truth, both in absolute and relative numbers:

		True Class	
		Re-offend	Not re-offend
COMPAS predictions	Re-offend	2035 (28.21%)	1282 (17.77%)
	Not re-offend	1216 (16.86%)	2681 (37.16%)

Table 38: The Confusion matrix for both races by Probublica [36], percentages sum to 100 pct. and represent total population.

		True Class	
		Re-offend	Not re-offend
COMPAS predictions	Re-offend	1369 (37.04%)	805 (21.78%)
	Not re-offend	532 (14.39%)	990 (26.79%)

Table 39: The Confusion matrix for African-American by Probublica [36], percentages sum to 100 pct. and represent total African-American population.

		True Class	
		Re-offend	Not re-offend
COMPAS predictions	Re-offend	505 (20.58%)	349 (14.22%)
	Not re-offend	461 (18.79%)	1139 (46.41%)

Table 40: The Confusion matrix for Caucasian by Probublica [36], percentages sum to 100 pct. and represent total Caucasian population.

As can be seen in the tables, the classifications of the COMPAS algorithm have more false positives for African-Americans (21.78%) than for Caucasians (14.22%), and that there are more false negatives for Caucasians (18.79%) than for African-Americans (14.39%). This shows a racially dependent bias.

The data can be found at ProPublica’s github [64].

I COMPAS - Descriptive Analysis

In the following section, an initial descriptive data analysis is performed on the COMPAS data set to better understand the data used for building the COMPAS algorithm and the models that should reproduce the COMPAS algorithm.

ProPublica tested racial disparities by creating a logistic regression model. This model considered race, age, criminal history, future recidivism, charge degree, gender, and age. [36]. The descriptions are known before the scoring-process, and could potentially improve the performance of the model - therefore, the charge descriptions are also included in the following initial descriptive data analysis.

The following columns are included:

Variable	Description	Type
sex	The sex of the defendant: female or male	String
age	The age of the defendant	Integer
age_cat	Three age categories (intervals)	String
race	Five different race categories and a category "other"	String
juv_fel_count	The number of juvenile felonies	String
juv_misd_count	The number of juvenile misdemeanors	Integer
juv_other_count	The number of prior juvenile convictions that are not counted in the two variables above	Integer
priors_count	The number of prior crimes comitted	Integer
c_charge_desc	A short (one line) description of the charge	String
c_charge_degree	Degree of the charge (felony or misdemeanor)	String
decile_score	The decile of the COMPAS score	Integer
is_recid	1, if the defendant did any criminal offense that resulted in a jail booking after the crime for which the person was COMPAS scored, else 0	Integer / Binary

Table 41: Description of selected COMPAS data used for classification

Sex
A defendant is categorized as either a female or male. Most of the defendants are males which constitues approx. 80% of the defendants in the dataset, where females constitutes the remaining 20%. This means that the majority of the defendants are males.

sex	count	percentage
Female	1395	19.34
Male	5819	80.66

Table 42: Count of records grouped by sex

Race
There are five unique race-categories and a category called "other" for those who were not in the five categories. This makes up six categories: 'African-American', 'Asian', 'Caucasian', 'Hispanic', 'Native American' and 'Other'. The number of records grouped by race are:

race	count	percentage
African-American	3696	51.23
Caucasian	2454	34.02
Hispanic	637	8.83
Other	377	5.23
Asian	32	0.44
Native American	18	0.25

Table 43: Count of records grouped by race

From table 43 the majority of the defendants are "African-American", 51%, and the subsequently largest category is "Caucasian" with 34%. Both "Hispanic" and "Other" represent respectively under 10% of the data set. "Asian" and "Native American" both represent under 1% respectively. This means that African-Americans are over-represented compared to the other races.

Looking across sex and race in table 44 the five most represented groups are male African-Americans ($\sim$ 42%), male Caucasians ($\sim$ 26%), female African-Americans ($\sim$ 9%), female Caucasians ($\sim$ 8%) and male Hispanics ($\sim$ 7%).

sex	race	count	percentage
Female	African-American	652	9.04
	Asian	2	0.03
	Caucasian	567	7.86
	Hispanic	103	1.43
	Native American	4	0.06
	Other	67	0.93
Male	African-American	3044	42.20
	Asian	30	0.42
	Caucasian	1887	26.16
	Hispanic	534	7.40
	Native American	14	0.19
	Other	310	4.30

Table 44: Count of records grouped by sex and race

Age and age categories
Figure 53 shows a right-skewed age distribution implying that the defendants tend to younger. The same conclusion is made when investigating the age distribution across both sex and race. The youngest defendant is 18, and the oldest is 96.

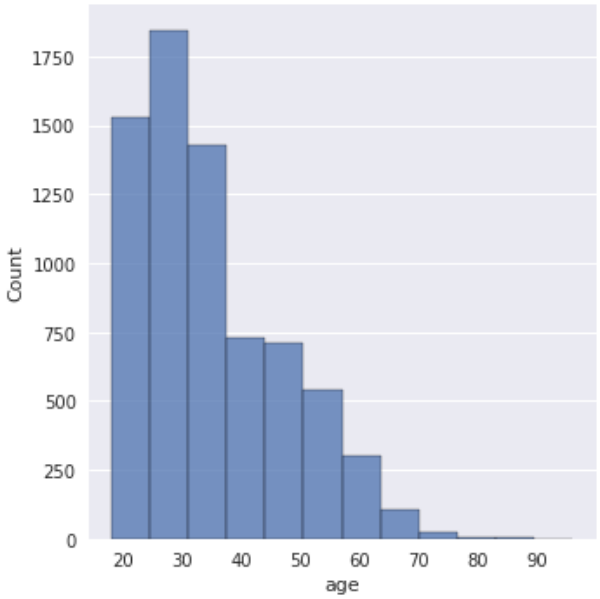


Figure 53: Histogram of the defendants' ages

The histogram in figure 53 is reflected in table 45 where almost 57 % of the defendants are between 25 and 45 years old.

age_cat	count	percentage
Less than 25	1529	21.19
25 - 45	4109	56.96
Greater than 45	1576	21.85

Table 45: Count of records grouped by age category

One should keep in mind, that the ages of the defendants range from 18 to 96, also, the interval sizes in age_cat are not consistent. The category "less than 25" spans over seven years, "25-45" spans over 20 years and "greater than 45" spans over 50 years. If the categories' span were divided into equal large sizes, as shown in figure 46, the results implies that most of the defendants are under 45 years old.

Updated age categories	count	percentage
Less than 45	5527	76.61
45-70	1652	22.90
Greater than 70	35	0.49

Table 46: New age categories where the age range are divided into three equal sized intervals

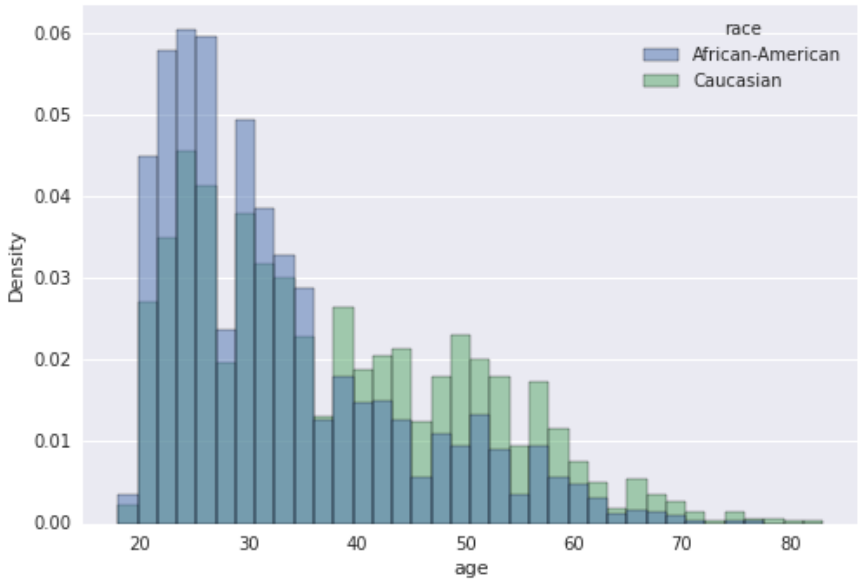


Figure 54: Histogram of the defendants' ages grouped by race

Juvenile convictions and prior crimes
97% of the defendants have done no juvenile felonies. Of the remaining 3%, 282 defendants, 189 have done a single juvenile crime. The defendant with the most juvenile felonies has committed 20 felonies. 95% of the defendants have no juvenile misdemeanors, and the defendants with the most juvenile misdemeanors have 13. 92% of the defendants have no other juvenile convictions, and the defendant with the highest number of other convictions is counting 17. 86.51% of the defendants have no juvenile felonies, no juvenile misdemeanors, and no other juvenile convictions.

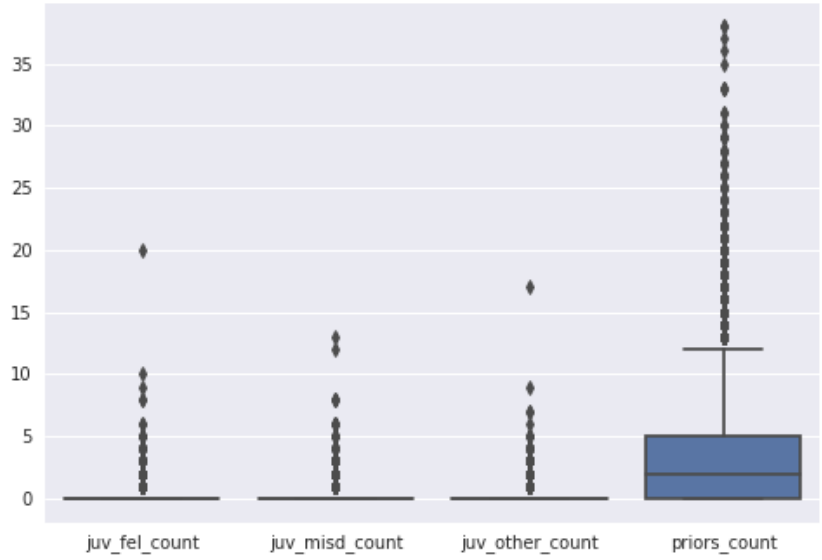


Figure 55: Boxplot of juvenile conviction variables and prior crimes.

Figure 55 shows that 50% of the defandants have less than 2 prior crimes. 25% have done 5 prior crimes

or more. The defendant with most prior crimes counts 38 crimes.

Charge degree and description
The charge degree is either "F" (felony) or "M" (misdemeanor). A misdemeanor is a less serious crime than a felony. Felonies are the most serious crimes one can commit, and they are associated with long jail or prison sentences. Misdemeanors often involve jail time, smaller fines and temporary punishments [65]. 4666 ($\sim$ 65%) of the records are felonies, and 2548 ($\sim$ 35%) are misdemeanors.

There are 437 unique charge descriptions each description is between one and eight words long. 50% of the descriptions are only used once. 16 ($\sim$ 4%) of the descriptions are each used for 1% of the records or more. The top five most frequent charge descriptions are shown in table 47 and the respective fraction (in percentage) of the total number of records in the data set.

c_charge_desc	count	percentage
Battery	1156	16.09
arrest case no charge	1137	15.82
Possession of Cocaine	474	6.60
Grand Theft in the 3rd Degree	425	5.92
Driving While License Revoked	200	2.78

Table 47: Top five most frequent charge description types.

Recidivism and decile scores
In figure 56 a histogram of the binary recidivism variable is plotted. 3743 (51.89%) defendants in the data set did not re-offend, and 3471 (48.11%) did re-offend.

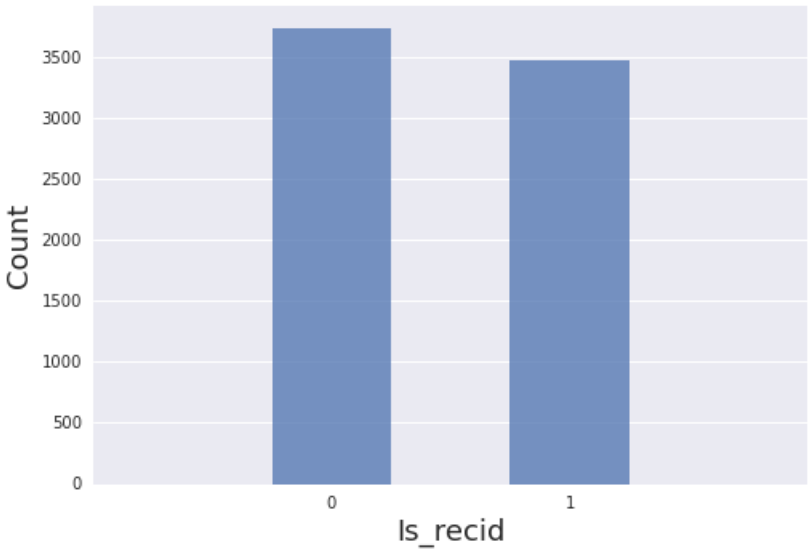


Figure 56: Histogram of recidivism

COMPAS decile scores are risk scores (integer type) ranged between 1 and 10, where 1 is the lowest risk score, and 10 is the highest risk score. 1-4 is labeled as "LOW", 5-7 as "MEDIUM" and 8-10 as "HIGH" [36]. Figure 57 shows a right-skewed distribution of COMPASS scores across all records in the data set.

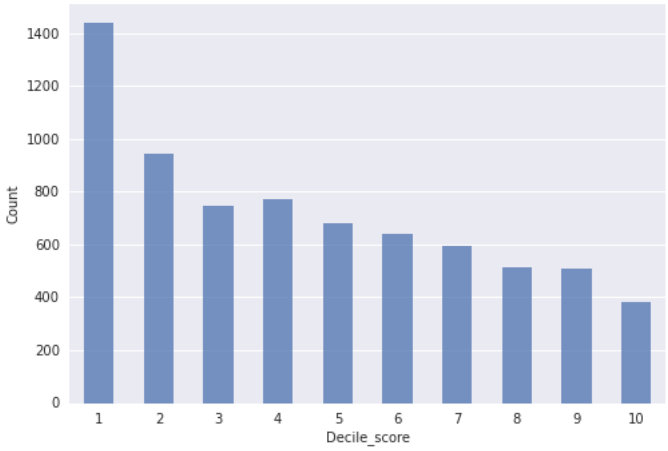


Figure 57: Histogram of decile COMPAS scores

ProPublica defines a binary recidivism variable as 0 if the decile score is under 5, else 1. Each of the COMPAS binary recidivism scores is shown as a histogram in figure 58, where the actual is_recid count is visualized as the red horizontal lines.

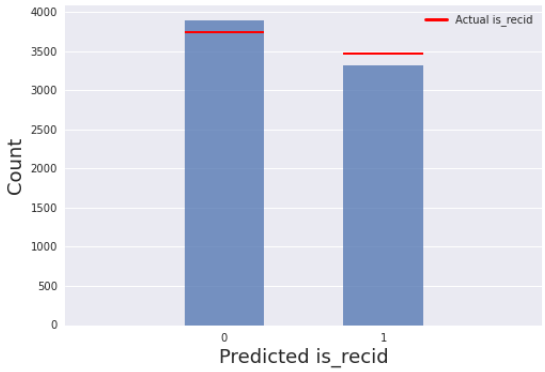


Figure 58: Histogram of predicted recidivism converted from COMPAS scores. The red lines show the counts from figure 56

Figure 58 shows a distribution that visually mimic the actual recidivism distribution from figure 56. The COMPAS predictions increase the distance between the two bars, which means that more defendants are predicted as not re-offending, even though they are, and vice versa with defendants that do re-offend. Since the classification problem is a supervised learning problem, one can better understand the performance of the COMPAS algorithm by investigating the relation between the predicted class and the true class. The elements of the confusion matrix are shown in table 48. Furthermore, the accuracy of the COMPAS algorithm is 65.23%.

	TPR (%)	FPR (%)	TNR (%)	FNR (%)
All defendants	61.65	31.45	68.55	38.35

Table 48: TPR, FPR, TNR and FNR for COMPAS predictions of the actual recidivism

Table 48 shows that 61.65% of the times that the algorithm predicts a criminal re-offending - he or she actual re-offend. It also shows that the algorithms perform better in predicting citizens that do not re-offend - 68.55% of the time that the algorithm predicts that a citizen does not re-offend - the algorithm was correct.

To investigate bias related to protected variables, the confusion matrix for both African-Americans and Caucasians are specified in table 49.

Race	TPR (%)	FPR (%)	TNR (%)	FNR (%)
African-American	70.97	43.92	56.08	29.03
Caucasian	51.02	23.16	76.84	48.98

Table 49: TPR, FPR, TNR and FNR COMPAS predictions of the actual recidivism for respectively African-American and Caucasian

What is interesting to notice is that their metric-pairs FPR/FNR and TPR/TNR are skewed between the two races. The accuracy for African-American and Caucasians are respectively 64.29% and 66.05%. If the algorithm was only to be evaluated on accuracy as a performance measure, the algorithm performance for each class compared to each other seems immediately equal. The difference in FPR between the two race groups are shown in table 49. This relates to defendants that are predicted to re-offend but actually did not. The difference in FPR is approximately 20 pct. points. The difference in FNR is also approximately 20 pct. points. This metric relates to defendants that are predicted to not re-offend but actually did.

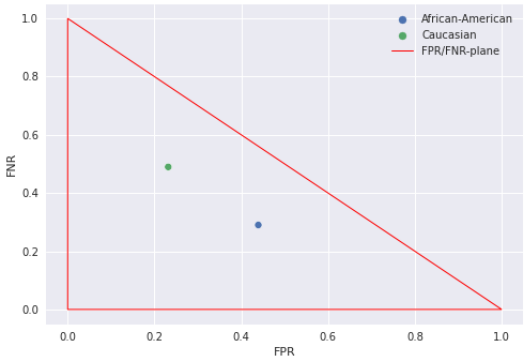


Figure 59: FPR/FNR plot for African-American and Caucasian

Figure 59 visualize the relation between FPR and FNR. Notice that "Caucasian" is located closest to the upper left corner of the triangle, which implies both lower FPR and higher FNR.

In figure 60 the decile score histogram for "Caucasian" shows a right skewed distribution - which means, that the distribution skews towards lower risk scores. The histogram for "African-American" shows a more uniform distribution. The caucasian distribution shows an especially high number of decile scores with the value 1. The appearance of the two groups' distributions is very different. Why the distributions form as they do can not be concluded based on the plots. However, potential reasons could be that, for example, the COMPAS algorithm is biased against African-Americans or, for example, African-Americans re-offend more.

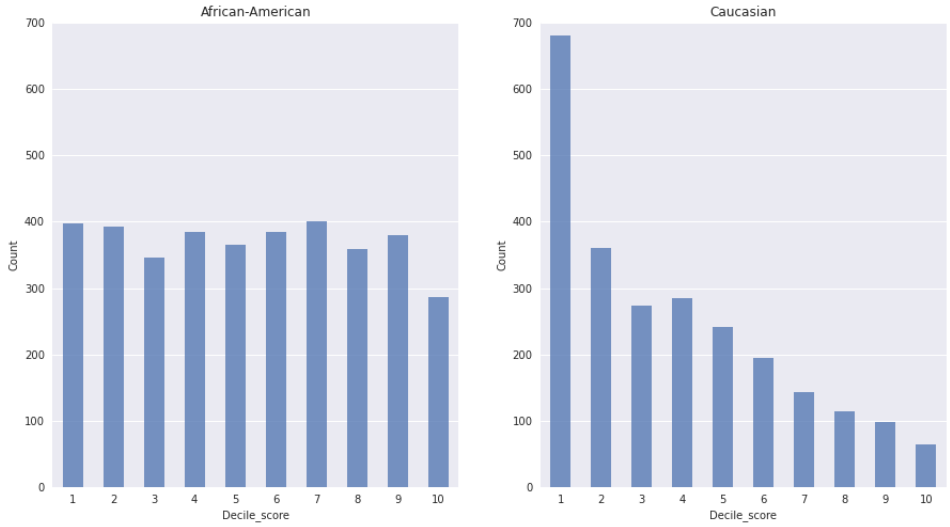


Figure 60: Decile score histogram for African-American and Caucasian.

Another potential protected variable in the COMPAS data set is the sex of the defendant. From table 42 it is shown that the majority of the defendants are males. Under-representation of a group can lead to performance issues, and since females are less represented than males, it would be interesting to investigate if there are any performance differences between the groups. The accuracy are 65.38% and 65.20% for females and males respectively - which shows that the algorithms performance equally "well" for both groups.

sex	TPR	FPR	TNR	FNR
Female	60.19	31.45	68.55	39.81
Male	61.92	31.45	68.55	38.08

Table 50: Performance of COMPAS predictions of the actual recidivism for respectively females and males

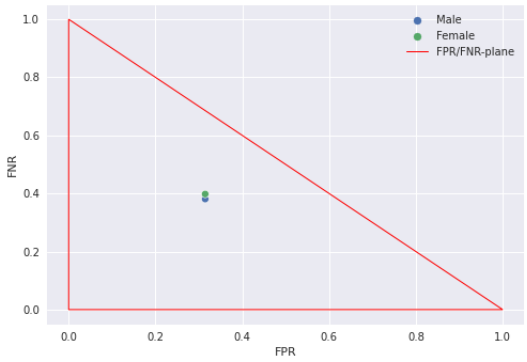


Figure 61: FPR/FNR plot for females and males.

Table 50 shows that across four metrics of the confusion matrix lies very close to each other. No obvious bias in terms of TPR, TNR, FPR and FNR. Also notice how close the two groups lie in figure 61, which implies that the COMPAS algortithm has almost no bias when comparing the scoring between

females and males. Yet, to better understand the distributions of the decile scores for each group the distributions are shown in figure 62.

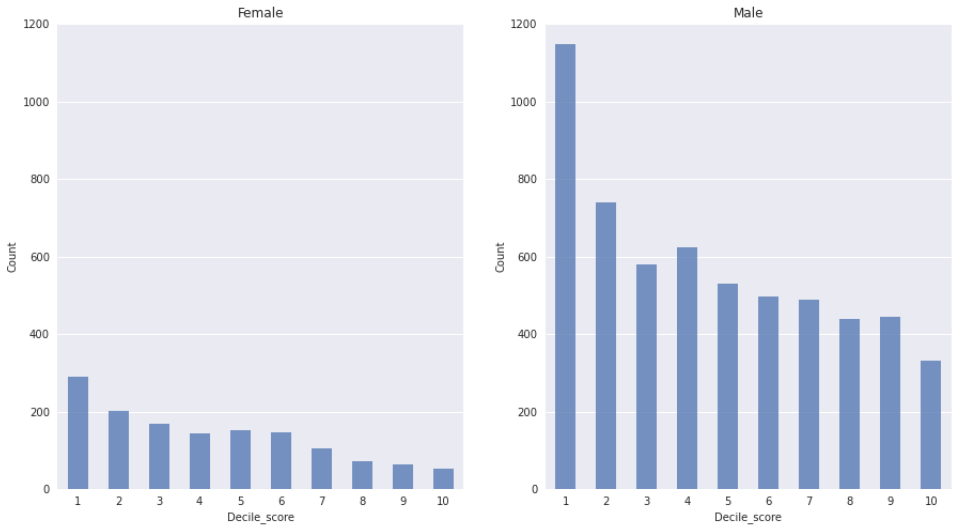


Figure 62: Decile score histogram for males and females

Both distributions are right-skewed, which means that both groups skew towards lower risk scores. The males have a high number of defendants with a decile score of 1, and the female distribution is a bit more flat (uniform).

As identified from table 44 there are differences in how much each of the sex-race groups makes up the total defendants in the data set. Since the amount of data for each sex-race group also could influence the performance of the COMPAS algorithm, this is further investigated.

sex and race	TPR	FPR	TNR	FNR
Male - African-American	71.15	45.33	54.67	28.85
Female - African-American	69.81	39.28	60.72	30.19
Male - Caucasian	49.88	20.82	79.18	50.12
Female - Caucasian	55.50	30.17	69.83	44.50

Table 51: Performance metrics of the COMPASS algorithm grouped by sex and race

Table 51 shows that regardless of the gender, FPR is higher for African-Americans, and FNR is lower for African-Americans - compared to Caucasians. In FPR, there are approximately ten percentage points in difference between caucasian males and females, where Caucasian males have the lowest FPR of all of the groups on 20.82 %. If one had only evaluated the COMPAS algorithm on accuracy, all four groups accuracy lies within the range between 64% and 67%.

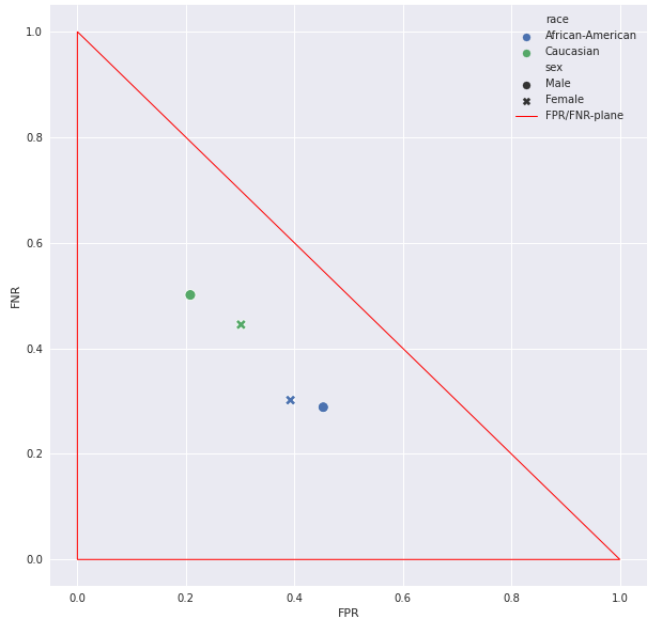


Figure 63: FPR/FNR plot grouped by sex and race

In figure 63 the caucasian groups both have lower FPR and higher FNR than the African-American groups. The Caucasian males are closest to the upper left corner of the triangle, which shows that this group has the most defendants with the lowest FPR and highest FNR. This means that male Caucasians are the group where defendants that do re-offend are not jailed - also, the group where the fewest number of defendants are jailed even though they will not recid.

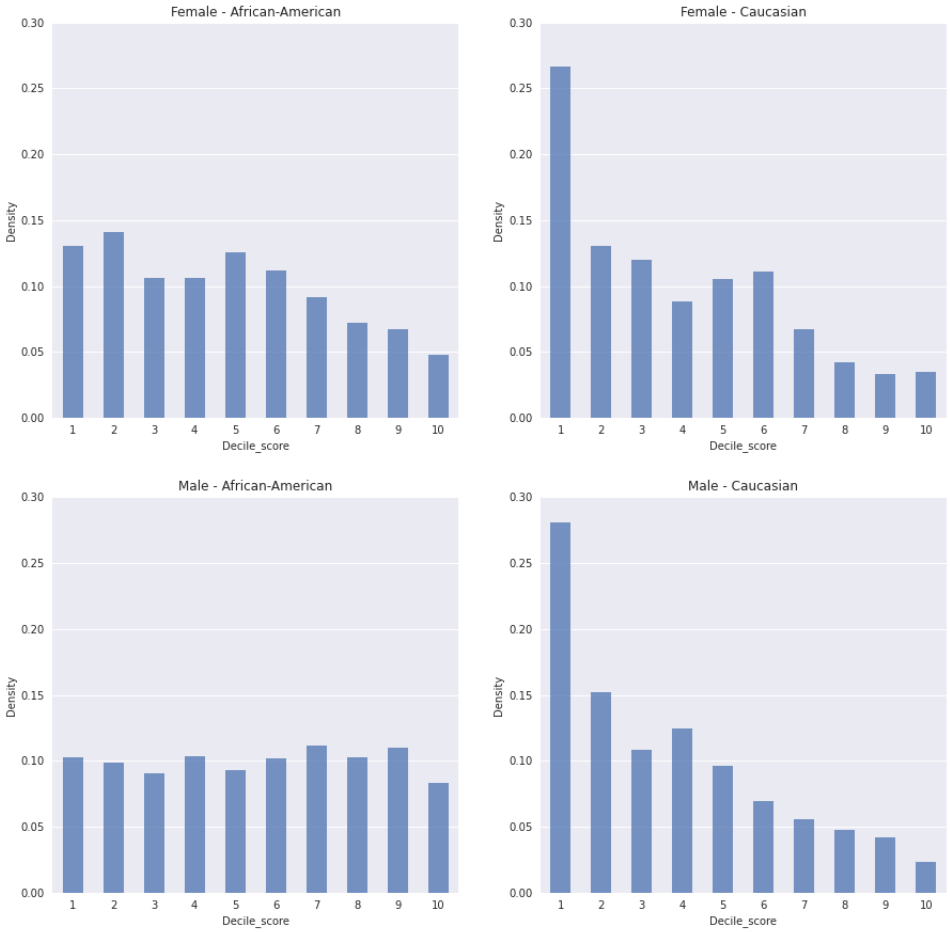


Figure 64: Density histogram grouped by sex and race

In figure 64 the density histograms show how both female and male Caucasians distributions are right-skewed. For female African-Americans there is a right-skew in the range between 5-10 in decile score, but it could also be argued to be a kind of bimodal with peaks at decile score 2 and 5 (and 1). The distribution for male African-Americans looks more uniform-like. For the groups with male African-Americans, a defendant is equally likely to get a score in the range between 1 and 10.

.

J COMPAS - Reproduction

Reproducing the COMPAS algorithm's bias

The COMPAS algorithm is a closed source commercial algorithm [7]. Therefore, assessing how techniques to identify and/or mitigate bias will affect the predictions is impossible since we do not have access to the model itself. We, therefore, attempt to create a model that reproduces the biases that have been highlighted as problematic in relation to the COMPAS algorithm. In their critique of the algorithm, ProPublica shows how *"blacks are almost twice as likely as whites to be labeled a higher risk but not actually re-offend"* and how the opposite is true for predictions regarding white defendants, where *"they are much more likely than blacks to be labeled lower-risk but go on to commit other crimes"* [7]. In other words, the algorithm predicts more false positives when assessing African-American defendants, and more false negatives when assessing Caucasian defendants.

By creating a model that reproduces the bias of the COMPAS algorithm, we wish to create a model on which we can test bias identification and mitigation techniques. The reasoning behind this is that if we build a model that reproduces the bias of interest found in the COMPAS algorithm, then hopefully, techniques that mitigate bias on our implementation might also work on the COMPAS algorithm.

To assess if our implementation reproduces the bias, we compare our model to the COMPAS algorithm in terms of the following classification measures: false positive rate (FPR), false-negative rate (FNR). These measures are included in ProPublica's investigation [7]. If our implementation resembles the COMPAS algorithm in terms of the classification measures presented above, we assess that we sucessfully have reproduced the COMPAS bias.

Our implementations

To reproduce the COMPAS bias in a robust fashion, we attempt to reproduce the COMPAS algorithm using four different models: a logistic regression, a support vector machine, a random forest, and a feed-forward neural network.
We use the models to predict recidivism among the defendants in the COMPAS dataset. As exogenous variables we include the following:

- age of the defendant
- race of the defendant (transformed to one-hot-encodings)
- sex of the defendant (transformed to one-hot-encodings)
- category of the charge (which led to the COMPAS screening) (transformed to dummy variables)
- Charge description transformed to one-hot-encodings)
- count of juvenile felonies
- count of juvenile misdemeanours
- count of prior offenses

Our implementations (that is: the logistic regression, the support vector machine, the random forest and the feed forward neural network) are all trained on a subset of the COMPAS dataset (training set) and tested on a testset. These results, along with the reported bias measures for the COMPAS algorithm found in [7], can be seen in table 52.

Test results: bias measures

By training the models on all defendants, reporting race-specific bias measures, and comparing the results to the bias measures of the COMPAS algorithm, we test whether or not we have succeeded in creating models that reproduce the bias of the COMPAS algorithm. The results are shown in figure 65.

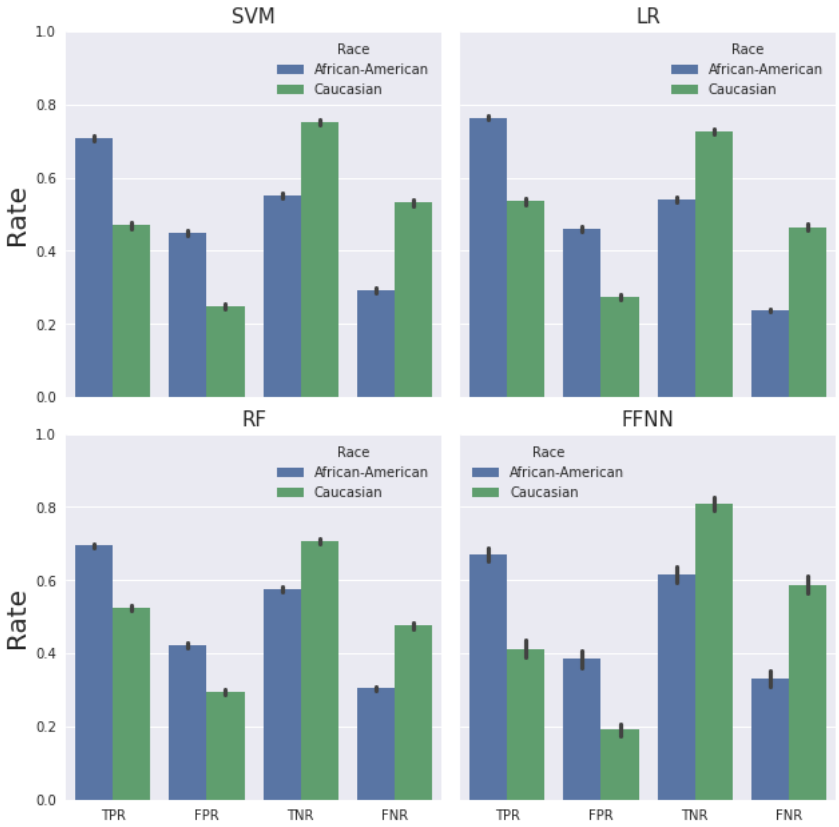


Figure 65: Bias measures from different models. A total overview of all bias measures for all models can be found in the appendix K.

From the figure it is clear that the models reproduce the most fundamental bias, that are found in the COMPAS algorithm, namely the differences in FPR and FNR between African-American and Caucasian defendants. Notice how each of the blue bars (African-American) in the TPR and FPR are much higher than the green bars (Caucasians). Also notice, how the opposite holds for TNR and FNR - where the green bars (Caucasians) are much higher than the blues (African-American). The COMPAS algorithm has a higher false-positive rate and a lower false-negative rate for African-American defendants and a lower false-positive rate and higher false-negative rate for Caucasian defendants. This central difference is reproduced in all of our implementations. This means, in other words, that if a judge were to follow the algorithmic recommendations, she could make two types of mistakes systematically: deny bail to African-American defendants who will, in fact, not re-offend and release Caucasian defendants who will re-offend. The estimates is found in appendix K table 52. All in all, we are satisfied with the results, which exhibits the same race-specific bias as the COMPAS algorithm.

K COMPAS - Reproduction metrics

Race	TPR	FPR	TNR	FNR
Support Vector Machine				
African-American:	**70.9**	**44.8**	**55.2**	**29.1**
	(70.2-71.6)	(44.1-45.6)	(54.4-0.6)	(28.4-29.8)
Caucasian:	**46.9**	**24.9**	**75.14**	**53.1**
	(45.9-47.8)	(24.0-25.7)	(74.3-76.0)	(52.2-54.1)
Logistic Regression				
African-American:	**76.3**	**46.0**	**54.0**	**23.7**
	(75.8-76.8)	(45.3-46.8)	(53.2-0.5)	(23.2-24.2)
Caucasian:	**53.6**	**27.3**	**72.69**	**46.4**
	(52.6-54.5)	(26.5-28.1)	(71.9-73.5)	(45.5-47.4)
Random Forest				
African-American:	**69.5**	**42.4**	**57.6**	**30.5**
	(68.9-70.1)	(41.6-43.1)	(56.9-0.6)	(29.9-31.1)
Caucasian:	**52.3**	**29.3**	**70.65**	**47.7**
	(51.5-53.2)	(28.6-30.1)	(69.9-71.4)	(46.8-48.5)
FFNN				
African-American	**67.0**	**38.5**	**61.5**	**33.0**
	(65.0-68.9)	(36.1-40.8)	(59.2-63.9)	(31.1-35.0)
Caucasian	**41.3**	**19.2**	**80.8**	**58.7**
	(38.8-43.7)	(17.5-20.9)	(79.1-82.5)	(56.3-61.2)

Table 52: COMPAS classification metrics of algorithms grouped by race

L Model architectures and hyperparameters

XGBoost

The following optimizer and hyperparameters has been chosen for AIR XGBoost:

- n_estimators: 400
- objective: "binary:logistic"
- scale_pos_weight : neg / pos. Where "neg" is the count of "No Falls" and "pos" is the count of "Falls"
- use_label_encoder: False
- learning_rate: 0.1
- eval_metric: "logloss"

Feed forward neural network

The following optimizer and hyperparameters has been chosen for AIR FFNN:

- Number of epochs: 400
- Number of nodes: 500
- Batch size: 40
- Optimizer: Adam
- Learning rate: 0.001
- Weight decay (L2-norm): 0.05
- Dropout rate: 0.4

The following optimizer and hyperparameters has been chosen for COMPAS FFNN:

- Number of epochs: 200
- Number of nodes: 2000
- Batch size: 40
- Optimizer: Adam
- Learning rate: 0.001
- Weight decay (L2-norm): 0.05
- Dropout rate: 0.5

The PyTorch model:

```
class Network(nn.Module):
    def __init__(self):
        super(Network, self).__init__()
        self.fully_connected1 = nn.Sequential(
            nn.Linear(n_feat,n_nodes),
            nn.ReLU(),
            nn.BatchNorm1d(n_nodes),
            nn.Dropout(p_drop)
            )

        self.fully_connected2 = nn.Sequential(
```

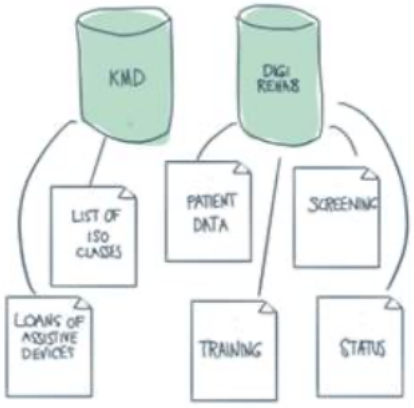
```
            nn.Linear(n_nodes,n_nodes),
            nn.ReLU(),
            nn.BatchNorm1d(n_nodes),
            nn.Dropout(p_drop)
            )

        self.fully_connected3 = nn.Sequential(
            nn.Linear(n_nodes,n_nodes),
            nn.ReLU(),
            nn.BatchNorm1d(n_nodes),
            nn.Dropout(p_drop)
            )

        self.fully_connected4 = nn.Sequential(
            nn.Linear(n_nodes,n_nodes),
            nn.ReLU(),
            nn.BatchNorm1d(n_nodes),
            nn.Dropout(p_drop)
            )
        self.fully_connected5 = nn.Sequential(
            nn.Linear(n_nodes,n_nodes),
            nn.ReLU(),
            nn.BatchNorm1d(n_nodes),
            nn.Dropout(p_drop)
            )


        self.fully_connected6 = nn.Sequential(
            nn.Linear(n_nodes,output_dim),
            nn.Sigmoid()

            )

    def forward(self, x):
        x = self.fully_connected1(x)
        x = self.fully_connected2(x)
        x = self.fully_connected3(x)
        x = self.fully_connected4(x)
        x = self.fully_connected5(x)
        x = self.fully_connected6(x)
        return x
```

M Internal AIR report

Christian Marius Lillelund
Research Assistant, MSc
Dept. of Electrical and Computer Engineering Aarhus University
Email: cl@ece.au.dk

" ... alle datasæt er indsamlet gennem virksomheden DigiRehab, der laver fysioterapeutisk genoptræning af borgere i Aalborg kommune, som har fået visiteret et hjælpemiddel (når en borger får visiteret et hjælpemiddel er kommunen pålagt at tilbyde træning jf. Serviceloven). DigiRehab har også fungeret som domæneeksperter i udforsknings- og træningsprocessen af data. De indsamlede data inkluderer følgende (se figur 3.1):"

- En liste af udlån af personlige hjælpemidler og deres ISO/HMI-numre af KMD.
- En liste med screeningsdata af DigiRehab, som fortæller, om den enkeltes borgers behov for hjælp og fysiske færdigheder. Disse er indsamlet ved fire ugers intervaller, hvis den enkelte borger harv æret i et rehabiliteringsforløb. I screeningsdata optræder ligeledes en liste med øvelser, som borgerne skal lave.
- En liste med træningsdata, som fortæller, hvornår enkelte borgere laver deres træning, om de laver den og hvordan deres træning er gået målt ved et subjektivt kriterie - SOSU-assistentens vurdering af intensitet/meningsfuldhed.
- En liste med faldobservationer registreret i Aalborg Kommune i systemet CURA. Disse fortæller, hvem der faldet, hvornår faldet er sket og andre omstændigheder, der har været ved faldet.



"Det er blevet besluttet at tage udgangspunkt i data fra DigiRehab i alle cases og til hver case udvikle en model, der givet en førstegangsscreening (baseline) kan forudsige, om en borger vil gennemføre sit træningsforløb eller ej, om en borger vil opnå compliance i sit træningsforløb eller ej, eller om en borger vil opleve et fald i sit forløb eller ej. Alle cases vil i første omgang udelukkende være baseret på generalia om borgeren (alder, køn) og oplysninger om borgerens hjælpemidler på det tidspunkt, hvor førstegangsscreeningen blev foretaget. Det er hensigten, at der i næste iteration af ML-delen skaludvikles yderligere en model, som givet en screening kan estimere en borgers faldrisiko i de næste tre måneder."

Her en liste over features, vi bruger for cases 1-3:

- Gender. En binær værdi, som indikerer borgerens køn.
- BirthYear. De sidste to cifre i borgerens fødselsdato.
- Cluster. En værdi mellem 0 og 32, som indikerer et tilhørsforhold en borgers hjælpemidler har med andre borgere. Hver værdi er i dette henseende en gruppe (cluster), og borgere i samme gruppe har udlånsmønstre af hjælpemidler, som minder om hinanden.

- NumberAts. Antal hjælpemidler, som borgeren har fået visiteret.
- LoanPeriod. Det gennemsnitlige antal dage, som borgeren har har lånt hjælpemidler af kommunen.
- 1Ats til 50Ats (50 kolonner). Lister med unikke ISO/HMI-numre på de hjælpemidler, som borgeren har fået visiteret sorteret efter visitationsdato. Medtaget er alle hjælpemidler (maksimalt 50).